

6502 • 80

NO. 12
1982
HK$6.00

Pac-Man 遊戲機大揭秘　　頂級 Games 程式 Listing

MICRO 83

將於82年12月中推出，其中的“101”BASIC集一向受讀者所歡迎。
全年的“101”BASIC集由JOSEPH君監製，JOSEPH君的作品已列出的包括有M.C.寫的太空怪，迷宮……等。

現代電子不惜巨資貢獻給各位的“101”BASIC 集

PLOTTING
AMAZING
ANIMALS
MODULO CONVERTION
DETERMINANT
EQUATION
PRIME NUMBER
QUADRATIC EQUATION
COIN GAME
CATION
BUNNY
TELEPHONE DIRECTORY
WRITE REMARKS
SPEED READING
CHECKER
HANGMAN
LOVE
COMPUTER MIND
THREE DIMENSIONAL GRAPH
MASTER MIND
IQ-TESTER
MEMORY CARD
SNOOPY
LAST STONE
REVERSE
CODING
ABBREVIATION LEARNING
TYPING SPEED
BOX GUESSING
MEN SAVE
REGRESSION
NUMBER REARRANGEMENT
SIMPSON'S APPROXIMATION
CUBIC EQUATION
STANDARD DISTRIBUTION
LAST CARD
JUMP
FOOK
HORSE RACING GAME
MAZE GENERATOR
PICK NUMBER
NUMBER GUESSING GAME
CREATE SHAPE TABLE
CALENDAR
PUZZLE I
EQUATION OF DEGREE 4
MEMORY
SENTENCE GUESSING
MARKS ADJUSTMENT
BLACKJACK
GRADE ALLOCATION
DIE GAME
SIN/COS/TAN FUNCTION
CALCULATOR
PROBABILITY
TIME TABLE
OHM'S LAW
ARC OF SIN/COS/TAN
THREE KINGS
CRAZY COMPUTER
CALCULATION OF AREA
LOGARITHM
CALCULATION OF VOLUME
MEMORANDA ORIGINAL
SHOOTING
EARTH BOMB
MID$ SUBROUTINE
PI
TRINOMIAL
BINOMIAL
BIRTHDAY REMINDER ORIGINAL
LETTER GAME I
NAME SORTING
APPLECATCHER
INSTR SUBROUTINE
POLYNOMIAL OPERATION
LETTER GAME II
FALLING OBJECTS
THREE DIGITS
TARGET
VOCABULARY LEARNING ORIGINAL
FOOL FIRING GAME
LOVE STORY
LYING
SIMPLE ALGEBRA
MESSAGE CODING/DECODING
SIMPLE ARITHMETIC
GHOST
COLOR CODE DECODER
REGULAR POLYGONS
TRAFFIC
MG 880
CROSS THE RIVER
DATE-OF-BIRTH

101簡目

MICRO 83 出版
定必轟動

電子科學出版社
INTERFACE ELECTRONICS
6502.80

i.e.
6502.80

STAMP
HERE

九龍新蒲崗衍慶街
衍慶大厦六樓B3室

訂閱通知書

茲訂閱 INTERFACE ELECTRONICS 6502·80 月刊 份 訂期：由第 期起

寄法：☐平郵 ☐空郵 ☐掛號

☐附上劃綫支票 No. ________________

＊＊支票收款人請寫"INTERFACE ELECTRONICS PUBLISHERS"

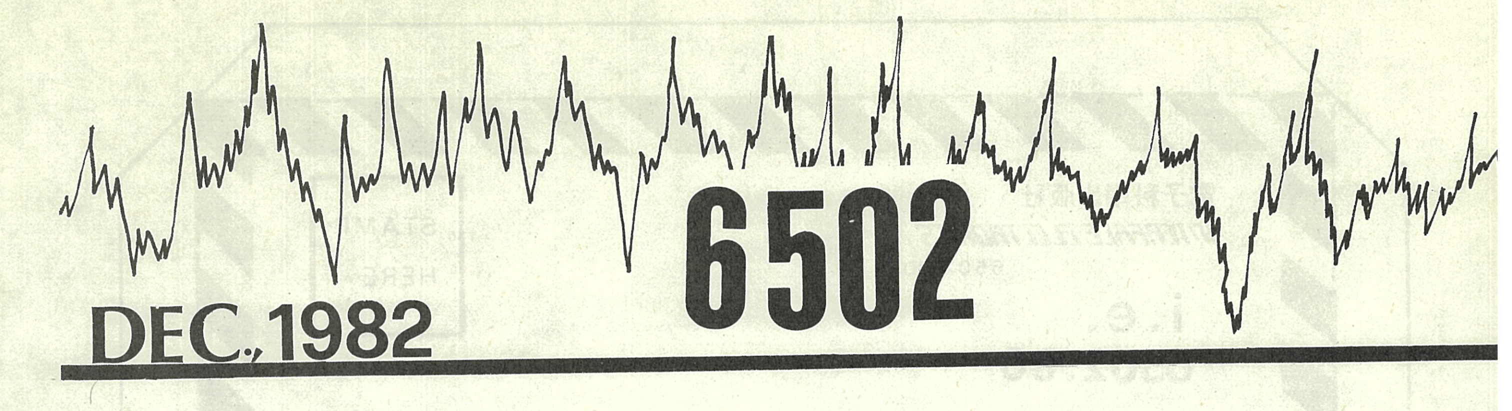
訂户姓名________________電話____________

地址________________________

（外埠訂戶請用英文大楷）________________

全年十二期訂閱費

香港 HK $72.00
外埠 HK $140.00
空郵東南亞地區 HK $190.00
其他地區 HK $240.00
（掛號另加 HK $48）



目錄

督印： 陳金燿
主編： 羅偉民 80
吳偉聰 6502
執行編輯： 湛婉貞
黃惠芳
封面設計： 陸子峯
承印：堅利印刷有限公司
發行：同德書報社

80

通訊地址：

INTERFACE ELECTRONICS PUBLISHERS

九龍新蒲崗衍慶街衍慶大厦六樓B3室 *TEL-237644*



一般電子遊戲中心都常見，現在將這遊戲形式搬到VIC－20裏運行，但要加了KRM，EXP－NDER，因爲程式要EXPA－NDER的RJOY（0）同3KRAM。

遊戲開始，一隻青蛙在街一邊，要去河一邊入洞取分數，在街上有很多車來往，青蛙要避開直到河邊，跳上河上的浮木，利用浮木跳到河一邊入洞，青蛙可以八個方向行動，即上下左右，或45°向上左，上 、下左，下右，青蛙向前行一步可得十分，入了洞可得二十分，但向後行一步便減十分，如途中給車碰倒，跳入河中，跳不正洞或三十秒之內入不到洞便損失青蛙一隻，當青蛙入了5隻之後河上的浮木便將長度減少，增加難度，如程式效果不滿意，可看下文自行修改。

LINE130GOTO1120是將字母砌成車、木頭、青蛙。

LINE140至1270印出畫面的形式，如青蛙隻數少或分數少可在LINE160的B＝3或N＝10加大，B是青蛙隻數，N是分數。

LINE280至1000是程式中心，LINE310至320是將浮在水中的木頭轉動，做到木頭在中一樣浮動。

LINE570至610是將間定在三十秒，如三十秒不足或三十秒多可以將LINE430同LINE570的TI$ =>"000030"修改來增加難度。

```
100 REM *************
110 REM *STREET FROG*
120 REM *************
130 GOTO1120
140 POKE36869,255:POKE36879,30
150 C=0:D=100:SC=0:F=8152:A=0:OF=30720+F
160 B=3:N=10:TI$="000000"
170 Z$="▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓"
180 PRINT"▓":POKEA+F,0
190 A$="DDDDDDDDDDD           "
200 B$="DDDD   DDDD   DDDD    "
210 C$="DDDDDD     DDDDDD     "
220 D$="DDD        DDD        "
230 E$="ABC ABC   ABC ABC     "
240 F$="EBF      EBF      EBF "
250 G$="ABC     ABC     ABC ABC "
260 FORMM=8164TO8185:POKEMM,7:POKEMM+30720,6:NEXT
270 PRINTZ$;"▓▓B.B. CLUB▓"
280 FORI=0TO21:POKE36878,15:POKE36877,129
290 PRINT"▓▓▓▓▓▓▓▓▓▓▓▓▓▓GGGGGGGGGGGGGGGGGGGGGG"
300 PRINT"▓▓▓G▓GGGGG▓GGGGG▓GGGGGG▓G"
310 CR=PEEK(7200)
320 FORMM=7200TO7206:POKEMM,PEEK(MM+1):NEXT:POKE7207,CR
330 PRINT"▓";LEFT$(Z$,4)RIGHT$(A$,I)LEFT$(A$,22-I)
340 PRINT"▓";LEFT$(Z$,6)RIGHT$(B$,22-I)LEFT$(B$,I)
350 PRINT"▓";LEFT$(Z$,8)RIGHT$(C$,I)LEFT$(C$,22-I)
360 PRINT"▓";LEFT$(Z$,10)RIGHT$(D$,22-I)LEFT$(D$,I)
370 PRINT"▓";LEFT$(Z$,14)RIGHT$(E$,I)LEFT$(E$,22-I)
380 PRINT"▓";LEFT$(Z$,16)RIGHT$(F$,22-I)LEFT$(F$,I)
390 PRINT"▓";LEFT$(Z$,18)RIGHT$(G$,I)LEFT$(G$,22-I)
400 PRINT"▓▓SCORE"SCTAB(12)TI$"▓▓▓▓▓▓▓TIME▓"
410 IFPEEK(A+F)=4ANDC>5THENGOSUB570
420 IFPEEK(A+F)=32ANDC>5THENGOSUB670
430 IFTI$=>"000030"THENGOSUB620
440 IFC<5THENGOSUB720
450 POKEA+F,32
460 IFC=10THENGOSUB780
470 L=RJOY(0)
480 IFLAND1THENA=A-44:C=C+1:SC=SC+10:BB=240
490 IFLAND2THENA=A+44:C=C-1:SC=SC-10:BB=240
500 IFLAND4THENA=A-1:N=N-1:BB=240:IFN=0THENGOSUB620
510 IFLAND8THENA=A+1:N=N+1:BB=240:IFN=21THENGOSUB620
520 IFA>0THENA=0
530 POKEA+F,0:POKEOF+A,5:POKE36876,BB
540 FORU=1TO50:NEXT:POKE36876,0 :BB=0
550 NEXT
560 GOTO280
570 IFTI$=>"000030"THENGOSUB620
580 IFC=6THENA=A-1:N=N-1:RETURN
590 IFC=7THENA=A+1:N=N+1:RETURN
600 IFC=8THENA=A-1:N=N-1:RETURN
610 IFC=9THENA=A+1:N=N+1:RETURN
620 REM
630 POKE36877,220:FORL=15TO0STEP-1:POKE36878,L:FORU=1TO100:NEXTU,L
640 POKE36878,0:POKE36877,0:B=B-1:POKEA+F,32
650 IFB=0THEN1010
660 A=0:C=0:N=10:TI$="000000":RETURN
670 POKE36877,252:FORL=15TO0STEP-1
```

ＬＩＮＥ８００至９３０是青蛙入３洞超個五隻便將浮在水上的木頭長度減短。

ＬＩＮＥ１０１０至１１１０是遊戲結束，記着最高記錄，同時問斷續遊戲嗎。如果不斷續按鍵盤的Ｎ字，如斷續按鍵盤的Ｙ字或ＪＯＹ－ＳＴＩＣＫ的ＦＩＲＥ制便可。

```
680 POKE36878,L:FORU=1TO90:NEXTU,L
690 POKE36878,0:POKE36877,0
700 A=0:C=0:N=10:TI$="000000":B=B-1:IFB=0THEN1010
710 RETURN
720 IFPEEK(A+F)<>32THEN740
730 RETURN
740 IFPEEK(A+F)=32THENRETURN
750 IFPEEK(A+F)=0THENRETURN
760 GOSUB620
770 RETURN
780 POKE36877,0:IFA+F=7703ORA+F=7709ORA+F=7712ORA+F=7715ORA+F=7722THEN800
790 GOTO990
600 X=X+1:IFX=5THEND=D-20:IFD=80THEND$="    DDD
810 IFD=60THENB$="          DD      DD     DD"
820 IFD=40THENC$="        DDDD    DDDD      "
830 IFD=20THENA$="      DDD    DDD      DDD  "
840 IFD=0THEND$=" DD      DD      DD      DD "
850 IFD<0THEND=0
860 POKEA+F,0
870 SC=SC+20
680 POKE36878,15:FORL=1TO15
890 FORM=200TO220+L*2:POKE36876,M
900 NEXTM,L:POKE36878,0:POKE36876,0
910 IFX=5THENGOSUB940
920 A=0:C=0:N=10:TI$="000000"
930 RETURN
940 SC=SC+50:X=0
950 POKE36878,15:FORL=130TO254:POKE36876,L
960 FORM=1TO40:NEXTM,L
970 POKE36878,0:POKE36876,0
980 RETURN
990 GOSUB620
1000 GOTO280
1010 POKE36879,27:SOUND0,0,0,0,0:POKE36869,240
1020 PRINT"▮▮ "
1030 PRINT"▮ ▮ ▮";"▮"SC
1040 PRINT"▮ ▮ ▮▮";"▮"SC
1050 PRINT"▮▮ "
1060 PRINT"▮▮▮▮▮YOUR SCORE IS▮"SC
1070 PRINT"▮▮▮ANOTHER GAME(Y/N)?"
1080 GETV$
1090 IFV$="Y"ORRJOY(0)AND128THENRUN140
1100 IFV$="N"THENPRINT"▮▮▮▮SYS64802▮▮▮":END
1110 GOTO1080
1120 POKE52,28:POKE56,28:POKE51,0:CLR:POKE37150,3
1130 K=7168:L=25600:FORI=K+256TOK+264:POKEI,PEEK(I+L):NEXT
1140 FORI=KTO7231:READJ:POKEI,J:NEXT
1150 DATA153,153,126,24,219,189,153,129,7,63 ,127,127,127,127,63 ,7
1160 DATA0,255,255,255,255,255,255,0,112,254,241,241,241,241,254,112
1170 DATA238,221,187,119,238,221,187,119
1180 DATA14,127,143,143,143,143,127,14,112,254,255,255,255,255,254,112
1190 DATA255,255,255,255,255,255,255,255,255
1200 GOTO140
0 "▮▮▮
0 "▮▮▮   B.B. CLUB
```

坦克大決戰

坦克大決戰——VIC—20

這個來自I、E65φ2編輯部的遊戲程式，名爲坦大決戰。它是由兩個人一起準行遊戲。遊戲者的目標就是要在對方未摧毀自己的坦克前，把敵人摧毀。

遊戲者利用以下的鍵來移動坦克的位置：

左方

E

S　D“3”發炮

X

右方

@

:　;　“—”發炮

/

當遊戲進行的時候，你會發現螢幕上會出現一些細小的綠點。不要輕視這些點，這些點就是坦克在地下的地雷。這些地雷非常厲害，甚至炮彈也不能把它們引爆。但是如果你的坦克撞中它們，你的坦克便會爆炸，而你的對手便會得到一分。

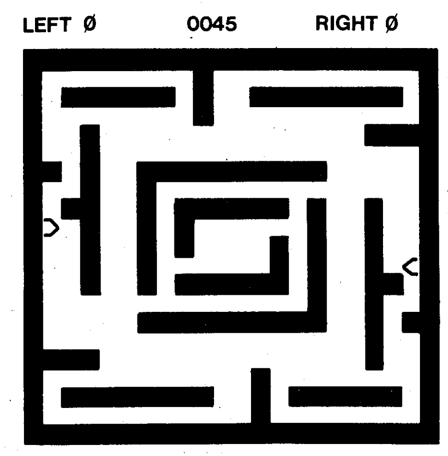
另一個勝利的方法，（得分的方法）就是把你的敵人摧毀。當你按動發炮掣的時候，一顆炮彈便會從你最後移動的方向發射。如果炮彈擊中牆壁，你可以等待炮彈消失後，再次發炮。換句話說，在發炮途中，你不可以繼續發炮。

這個遊戲是有兩個版本的。上文所述的就是遊戲B的大概情況；在遊戲A中，任何一位遊戲者按動發炮掣的時候，電腦便會在螢幕上隨意（random）繪出一件方塊。過了一段時間後，整個螢幕可能佈滿了地雷和方塊。

如果遊戲者要記著以前的分數，又要重新開始遊戲，可以按f 1鍵。按動此鍵可以容許你跳往遊戲A或遊戲B，而沒有失去先前的紀錄。

在螢幕上方的數字就是代表時間，當某輛坦克被摧毀後，時計便會停止。

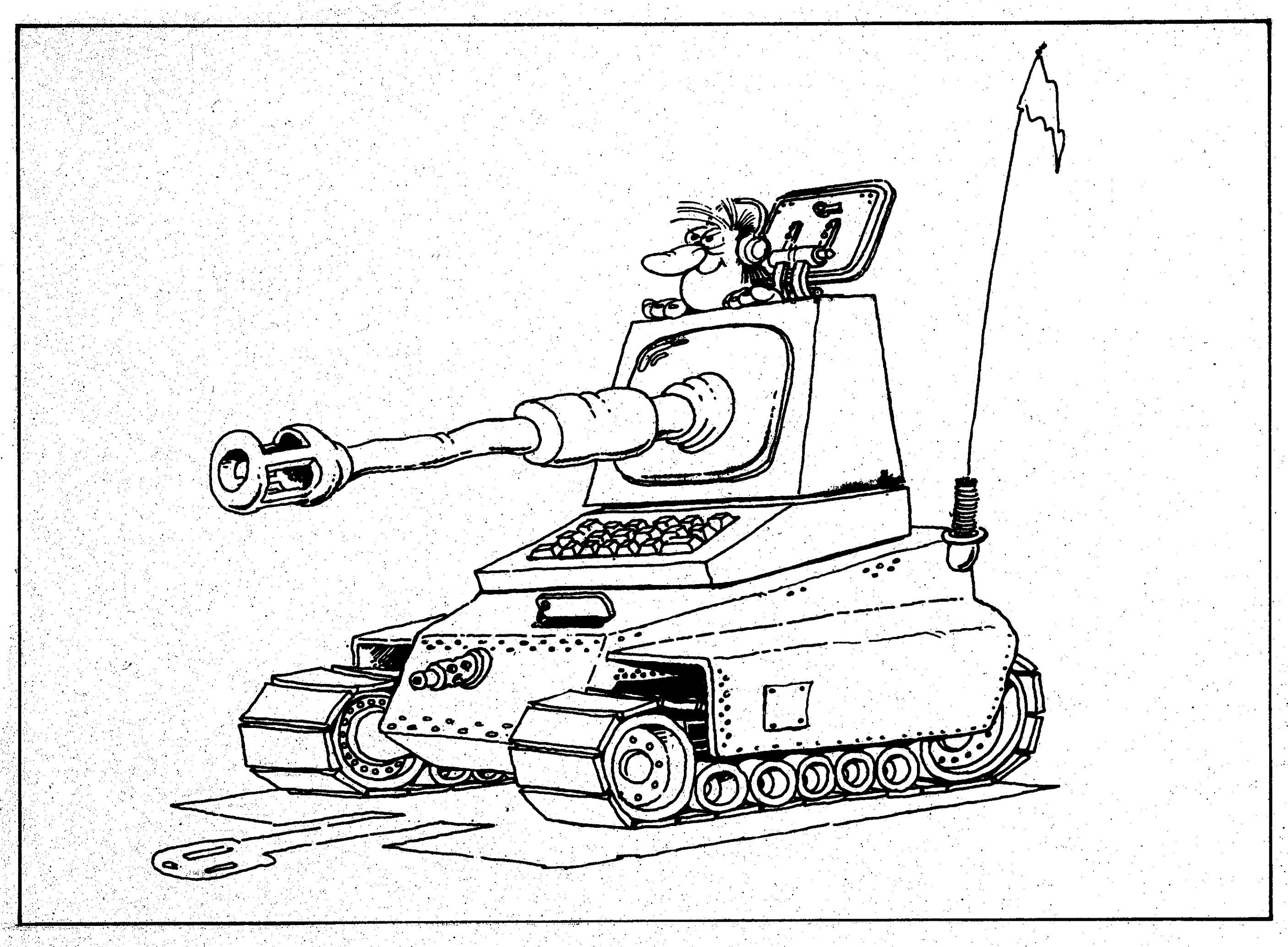
當運算（Run）的時候，程式便產生一個迷宮（看圖一）



圖一

（假定應用同樣的數據）。>符號就是代表左方遊戲者的坦克，而<符號就代表右方的坦克。而在左右方隨着“LEFT”或“RIGHT”所顯示的數字就是代表他們所得的分數。

當每次遊戲完畢的時候，最底



的一行便會顯示誰是優勝者，而第二行便會出現問題；"Anoth－er go（Y／N）"。

行002：這行定出其他程式部份在螢幕裏的顏色。

行004：在這裏任何在GET語句裏所記着的鍵都會消去，被RESFT。除此之外，行4和行5也在螢幕的上端和下端負責產生一些彩色方塊。

行005至010：

這幾行負責輸出題目頁（TI－TLE PAGE）。注意行6和行8，在反白命令後是有13個空白位的。

011：這行設定變數。S0，S2和V都是來控制聲音效果的。S0和S2是兩個聲音盒子的記憶位置，而V是聲響（VOLUME）的控制。

015：注意這一行是不需分號（Semicolon）的。

016：要得到這個圖像符號，按下Commodore鍵，按下"Y"

017與019：

這兩行所出現的線形圖像符，要一起按動Commodore鍵和"T"才能夠得到。

026：這行利用邏輯保存着If then語句的一貫應用方法。如果GM被設定爲－1，那麼A$便等如"B"，否則便會等如"0"。

027：當遊戲者按下"f 1"鍵的時候，RESTORE便會作用。

040和050：

這兩行數據是用作繪畫迷宮的。從螢幕位置7724，或第三行以下，螢幕的號碼是由0至461。第一個數據號數就是指明從那一行開始，而第二個數據就是指行的號數。行40是所有水平線的數據，而行50是所有垂直線的數據。兩行最尾所出現的數據－1，是用來作控制閱讀數據完結之用。

070：在這裏，設定了螢幕位置和坦克的數值。W爲左方坦克的值；而B則是右方坦克的值。

098：這行檢測"f 1"鍵可有被按動。

380：在畫面閃動的時候（顏色轉變），這行輸出爆炸的聲響效果。

635：注意在浮標向下（Cur－sor down）指令後是有20個空白位的。

750：當在螢幕上隨意繪畫地雷後，顏色便會轉爲綠色。

這個程式的趣味性極濃，嘗試看看誰是坦克高手？

```
0 REM FROM I.E. 6502 EDITORIAL DEPARTMENT
1 REM TANK WARS
2 POKE 36879,42
4 POKE198,0:PRINT"[CLR][BLK]":FORA=0TO21:POKE7680+A,160:POKE38400+A,(ANND7):POKE8164+A,1
60
5 POKE38884+A,(AAND7):NEXT
6 PRINT"[DOWN][DOWN][DOWN]     [RVS]                ":PRINT "     [RVS] TANK BATTLE ":PRINT"     [RVS]
  "
7 PRINT"     [RVS]      FOR      ":PRINT"     [RVS]                ":PRINT"     [RVS]  V I C - 20 "
8 PRINT"     [RVS]                [DOWN][DOWN]":PRINT"      I.E.6502":PRINT:PRINT"EDITORIAL DEPARTM
ENT"
9 PRINT:PRINT"       TANK WARS [DOWN][DOWN]":PRINT"       [DOWN]HIT A KEY "
10 GETA$:IFA$=""THEN10
11 SO=36877:V=SO+1:S2=V-2:D1=-22:D2=D1
15 PRINT"[CLR]"TAB(5)"TANK BATTLE"
16 PRINTTAB(5)"---- -------[DOWN]"
17 PRINT"  THERE ARE TWO GAMES TO PLAY.A & B.[DOWN]":PRINT"GAME A":PRINT"------"
18 PRINT"THE COMPUTER WILL PUT UP OBSTRUCTIONS WHEN"
19 PRINT"THE 'FIRE' BUTTON IS PRESSED.[DOWN]":PRINT"GAME B":PRINT"------"
20 PRINT"SAME AS ABOVE BUT HAS NO OBSTRUCTIONS.[DOWN][DOWN][DOWN]"
22 PRINT"[RVS][RIGHT][RIGHT]PRESS 'A' OR 'B'":GOTO940
23 GETA$:IFA$=""THEN23
24 IFA$<"A"ORA$>"B"THEN22
26 GM=A$="B"
27 PRINT"[CLR]":RESTORE
28 READX,Y:IFX=-1THEN30
29 FORQ=XTOX+Y:POKE7724+Q,160:POKE8185-Q,160:NEXT:GOTO28
30 READX,Y:IFX=-1THEN40
32 FORQ=XTOX+22*YSTEP22:POKE7724+Q,160:POKE8185-Q,160:NEXT
33 GOTO 30
40 DATA 0,21,56,7,184,5,106,2,46,5,138,9,-1,-1
50 DATA 0,20,31,2,91,8,206,1,133,0,178,0,160,5,-1,-1
70 BC=60:WC=62:B=7986:W=7923::J=0:K=0:POKEB,BC:POKEW,WC
86 PRINT"[HOME]LEFT"ZZ"[RIGHT][RIGHT][RIGHT][RIGHT][RIGHT]RIGHT"ZX:TI$="000000"
90 GETA$:PRINT"[HOME][RVS][RIGHT][RIGHT][RIGHT][RIGHT][RIGHT][RIGHT][RIGHT][RIGHT]"RIGHT$(TI$,4)
91 IFK=1THENGOSUB220
93 IFA$="-"ANDJ<>1THENB1=B:J=1:D4=D2:GOSUB500:GOSUB400:GOSUB310:POKEB,BC
94 IFA$="3"ANDK<>1THENW1=W:K=1:D3=D1:GOSUB500:GOSUB400:GOSUB220:POKEW,WC
96 IFRND(1)>.98THENGOSUB750
97 IFA$=""THEN90
98 IFASC(A$)=133THEN4
100 IFA$="E"THEND1=-22:GOTO109
101 IFA$="@"THEND2=-22:GOTO112
102 IFA$="S"THEND1=-1:GOTO109
103 IFA$=":"THEND2=-1:GOTO112
104 IFA$="X"THEND1=22:GOTO109
105 IFA$="/"THEND2=22:GOTO112
106 IFA$="D"THEND1=1:GOTO109
107 IFA$=";"THEND2=1:GOTO112
108 GOTO 90
```

```
109 IFPEEK(W+D1)=32THENPOKEW,32:W=W+D1:POKEW,WC:GOTO90
110 IFPEEK(W+D1)=46THENPOKEW+D1,42:GOSUB800:GOTO356
111 GOTO90
112 IFPEEK(B+D2)=32THENPOKEB,32:B=B+D2:POKEB,BC:GOTO 90
113 IFPEEK(B+D2)=46THENPOKEB+D2,42:GOSUB800:GOTO355
114 GOTO90
220 IFPEEK(W1+D3)=32ORPEEK(W1+D3)=87THENPOKEW1,32:W1=W1+D3:POKEW1,81:RETURN
221 IFPEEK(W1+D3)=160ORPEEK(W1+D3)=46THENPOKEW1,32:K=0:RETURN
222 POKEW1,32:POKEW1+D3,81:GOTO355
310 IFPEEK(B1+D4)=32ORPEEK(B1+D4)=81THENPOKEB1,32:B1=B1+D4:POKEB1,87:RETURN
311 IFPEEK(B1+D4)=160ORPEEK(B1+D4)=46THENPOKEB1,32:J=0:RETURN
312 POKEB1,32:POKEB1+D4,87:GOTO356
355 X$=" LEFT":GOTO360
356 X$="RIGHT"
360 POKESO,220
380 FORL=15TO0STEP-1:POKEV,L:POKEV+1,L:FORQ=1TO300:NEXTQ,L:POKESO,0:POKEV+1,42
390 PRINT"▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓"X$" CONTROLLER WINS";
391 IFLEFT$(X$,1)="R"THENZX=ZX+1:GOTO600
392 ZZ=ZZ+1:GOTO600
400 POKESO,200:FORQ=1TO100:NEXT:POKESO,0:RETURN
500 IFOM=-1THENRETURN
510 TY=RND(1)*417+7746:IFPEEK(TY)=32THENPOKETY,160
530 RETURN
600 POKE198,0:PRINT"▓▓▓▓ANOTHER GO "
605 GOSUB900
610 GETA$:IFA$=""THEN610
620 IFA$="Y"THEN27
630 IFA$<>"N"THEN600
635 PRINT"▓▓                       ":FORQ=1TO500:NEXT
640 POKE198,0:PRINT"▓▓▓CHANGE GAME▓"
645 GOSUB900
650 GETA$:IFA$=""THEN650
660 IFA$="Y"THENRUN
670 IFA$<>"N"THEN640
680 END
750 TY=RND(1)*417+7746:IFPEEK(TY)=32THENPOKETY,46:POKE30720+TY,7
751 RETURN
800 FORL=1TO15:FORQ=200TO220+L*2:POKES2,Q:NEXTQ,L:POKES2,0:RETURN
900 POKEV,15:FORA=1TO5:PRINT"▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓Y▓N▓▓?":POKES2,225:FORQ=1TO150:NEXT
910 POKES2,0:PRINT"▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓(Y/N)":FORQ=1TO150:NEXTQ,A:RETURN
940 POKEV,15:FORA=1TO5:PRINT"▓▓▓▓▓▓▓▓▓▓▓A▓▓▓▓▓▓B":POKES2,255:FORQ=1TO150:NEXT:POK
ES2,0
950 PRINT"▓▓▓▓▓▓▓▓▓▓▓A▓▓▓▓▓▓B":FORQ=1TO150:NEXTQ,A:POKE198,0:PRINT"▓▓":GOTO23

READY.
```

VIC-20

超級迷宮



迷宮遊戲是很普通的遊戲，如果迷宮圖砌得精細也是相當不錯，現在ＶＩＣ　２０利用２２×２３來砌一個彩色迷宮，用ＪＯＹＳＴ－ＩＣＫ控制，而不需外加擴展。

遊戲方法很簡單，當你按ＲＵ－Ｎ時便開始砌圖，在砌圖其間看到一點方格向左右或上下移動，每移動一格便出現圖形，直致整個畫面完成及加了一層紅色便開始遊戲，在左上角有一計時器同你的位置，而出口在右下角，利用ＪＯＹＳ－ＴＩＣＫ以最快時間找尋出口，每當你行過的位置便加一層粉紅色代表你已經過，找到出口後遊戲便結束，同時顯示你的時間記錄，如果繼續遊戲便會出過新的迷宮圖，因爲迷宮圖是ＲＮＤ出來。

ＬＩＮＥ１０—５５清洗畫面同時加上黃色底。

ＬＩＮＥ６０—１０２５是整個迷宮圖形。ＬＩＮＥ１７０是ＲＮＤ出方格不同方向，做成不同劃面。

ＬＩＮＥ１０３０—１１００遊戲主程式。

ＬＩＮＥ１１１０—１８２０是ＧＡＭＥ，ＯＶＥＲ，ＬＩＮＥ１１２０—１１９０是音樂的ＦＯ－Ｒ　ＬＯＯＰ，如音響不合口味在ＬＩＮＥ２００００ＤＡＴＡ修改。

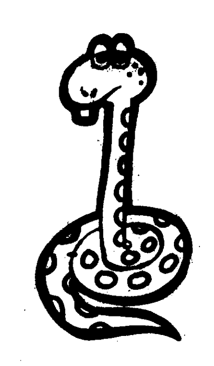
ＬＩＮＥ１００００—１００５０是一個副程式，每喚叫一次，便在畫面印出適當的牆壁。

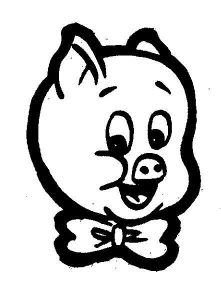
```
10 REM SUPER MAZE
20 POKE36879,93:POKE36878,7
30 PRINT"⊐":HS=PEEK(0)
40 XM=22:YM=21
50 DIMMA%(XM,YM),D%(3)
55 PRINT"⊐":FORL=38400TO38900:POKEL,7:NEXTL
60 FORL=1TO20:MA%(0,L)=10:MA%(XM,L)=10:NEXTL
70 FORL=1TO21:MA%(L,0)=5:MA%(L,YM)=5:NEXTL
80 MA%(0,0)=15:MA%(0,YM)=15:MA%(XM,0)=15:MA%(XM,YM)=15
95 X=10:Y=10
100 A=0
110 IFMA%(X-1,Y)=0THEND%(A)=1:A=A+1
120 IFMA%(X,Y-1)=0THEND%(A)=4:A=A+1
130 IFMA%(X,Y+1)=0THEND%(A)=2:A=A+1
140 IFMA%(X+1,Y)=0THEND%(A)=3:A=A+1
150 IFA=0ANDX=10ANDY=10THEN1000
160 IFA=0THENR=INT(MA%(X,Y)/256):GOTO230
170 B=INT(RND(1)*(A+1)):IFB>=ATHEN170
```

```
180 R=D%(B)
190 IFR=1THENMA%(X,Y)=(MA%(X,Y)OR1):MA%(X-1,Y)=(3*256OR4)
200 IFR=2THENMA%(X,Y)=(MA%(X,Y)OR2):MA%(X,Y+1)=(4*256OR8)
210 IFR=3THENMA%(X,Y)=(MA%(X,Y)OR4):MA%(X+1,Y)=(256OR1)
220 IFR=4THENMA%(X,Y)=(MA%(X,Y)OR8):MA%(X,Y-1)=(2*256OR2)
225 GOSUB10000
226 POKE38400+22*X+Y,7
230 IFR=1THENX=X-1
240 IFR=2THENY=Y+1
250 IFR=3THENX=X+1
260 IFR=4THENY=Y-1
265 GOSUB10000
266 POKE38400+22*X+Y,2
270 GOTO100
1000 X=1:Y=1:MA%(1,1)=MA%(1,1)OR8:GOSUB10000
1010 X=1:Y=0:MA%(1,0)=2:GOSUB10000
1015 X=1:Y=0:MA%(1,0)=2:GOSUB10000
1020 X=21:Y=20:MA%(21,20)=MA%(21,20)OR2:GOSUB10000
1025 X=21:Y=21:MA%(21,21)=8:GOSUB10000
1030 X=1:Y=0
1035 LET TI$="000000"
1040 POKE37139,128:L=PEEK(37137):POKE37154,127:R=PEEK(37152):POKE37154,255
1045 PRINT""INT(TI/6)/10
1046 POKE38400+22*X+Y,4
1050 IF(LAND4)<>4AND(MA%(X,Y)AND1)<>0THENX=X-1
1060 IF(LAND16)<>16AND(MA%(X,Y)AND8)<>0THENY=Y-1
1070 IF(RAND128)<>128AND(MA%(X,Y)AND2)<>0THENY=Y+1
1080 IF(LAND8)<>8AND(MA%(X,Y)AND4)<>0THENX=X+1
1090 POKE38400+22*X+Y,3
1100 IFX<>22ANDY<>21THEN1040
1110 RESTORE:Z=INT(TI/6)/10
1120 POKE36878,15
1130 FORL=0TO24
1140 READP,Q
1150 POKE36875,P:POKE36876,P
1160 FORT=0TO30*Q:NEXT
1170 POKE36875,0:POKE36876,0
1180 FORT=0TO10*Q:NEXTT
1190 NEXTL
1195 FORT=1TO2000:NEXT
1200 PRINT"YOU TIME IS";Z:PRINT"SEC."
1300 IFZ<HSTHENHS=Z:PRINT"YOU IS GOOD TIME"
1400 PRINT"GOOD TIME IS"HS
1700 PRINT" <GAME OVER>"
1800 PRINT"  HIT <RETURN>"
1810 GETA$:IFA$<>CHR$(13)THEN1810
1815 POKE0,INT(HS)
1820 RUN
10000 IF((MA%(X,Y)AND1)<>0)AND((MA%(X,Y)AND8)<>0)THENP=32+128:GOTO10040
10010 IF((MA%(X,Y)AND1)<>0)THENP=101+128:GOTO10040
10020 IF((MA%(X,Y)AND8)<>0)THENP=99+128:GOTO10040
10030 P=79+128
10040 POKE7680+22*X+Y,P:POKE36876,222:FORT=1TO5:NEXT:POKE36876,0
10050 RETURN
20000 DATA187,4,187,4,201,2,201,4,201,2,195,2,195,2,183,2,163,2,175,6,163,1,175,
1
20010 DATA183,2,195,2,195,2,201,2,195,2,183,2,163,3,175,1,183,2,183,2,175,2,175,
2,163,6
0 " B.B. CLUB
```

雅達利 Player Missile Graphic 初階

BY **BAN**





圖形移動的能力對每一個私人電腦都是重要的，它不但能增加遊戲的刺激性和吸引力，還提供一個生動的文字書述表達方式。

通常一般私人電腦產生圖形移動的方法是將整個圖形在原來的位置清除，然後在新的位置重新將整個圖形繪出，這個方法雖然行得通，但有一個弊病就是非常費時而且過程複雜。

雅達利私人電腦採用了一種名爲PLAYER／MISSILE GRAPHICS（P／M）來解決這個問題。P／M圖形是一種與普通GRAPHICS MODE圖形全不相同而且獨立的圖形，這就是說，它能與任何一個GRAP－HIC MODE同時存在而不影響其他圖形。

P／M圖形的記憶結構：（P－LAYER／MISSILE GRAPHICS MEMORY ORGANIZATION）

要知道P／M圖形是怎樣工作，我們先要明白P／M圖形的記憶結構。普通圖形之難以移動，是因爲畫面的結構是兩度空間而記憶中的結構是一度空間，每一個圖形都是由許多點組成，而每一點都在記憶中佔了一個位置，雖然這個圖形在畫面上是結合在一起，但在記憶中是非常分散的，移動這個圖形的時候，其實是將結合這個圖形的每一點清除然後再在新的位置繪出來，要將在記憶中那麼分散的東西適當地移動，必定要經過相當的計算，例如在BASIC，要移動一個圖形必定是用了很多PLOT，其實每一個PLOT都用了相當的計算方能在記憶中找到相應的點，更糟的是在滙編語言（ASSEMBLY LANGUAGE）的水平，這些計算還要親身做呢！

P／M圖形就是基於改善這個缺點而設計出來的，P／M圖形佔用記憶是一度高間，在畫面顯示出來亦是一度空間，P／M圖形的每一個PLAYER都佔用記憶中某256個BYTE，而這256個BYTE的資料就代表了一個PL－AYER的形狀。要知道，每個BYTE有8bits，每個bi－t能代表一點的點亮與否，那即是說，一個11111111 或FF 或255 的數便可代表一條由8點組成的線（圖1a）而一個10000001 或81 或129的數便可代表如圖1b的圖形。

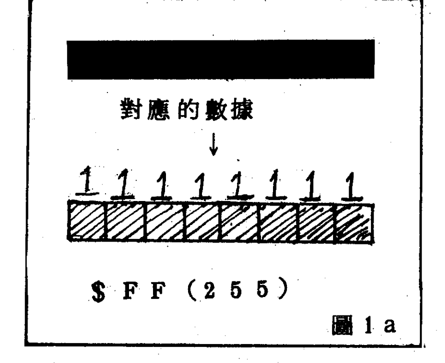


圖1a

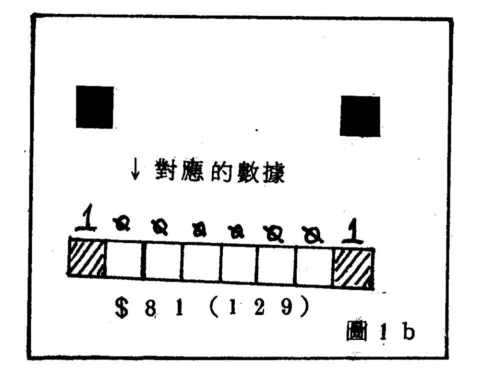


圖1b

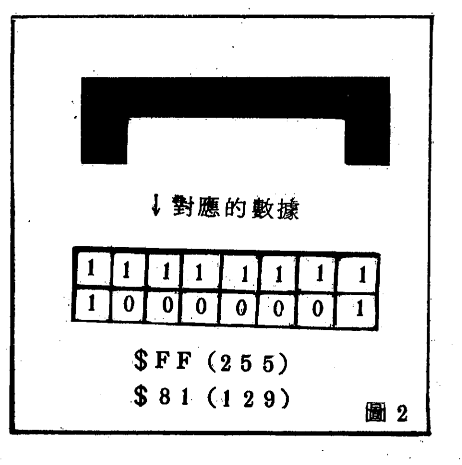


圖2

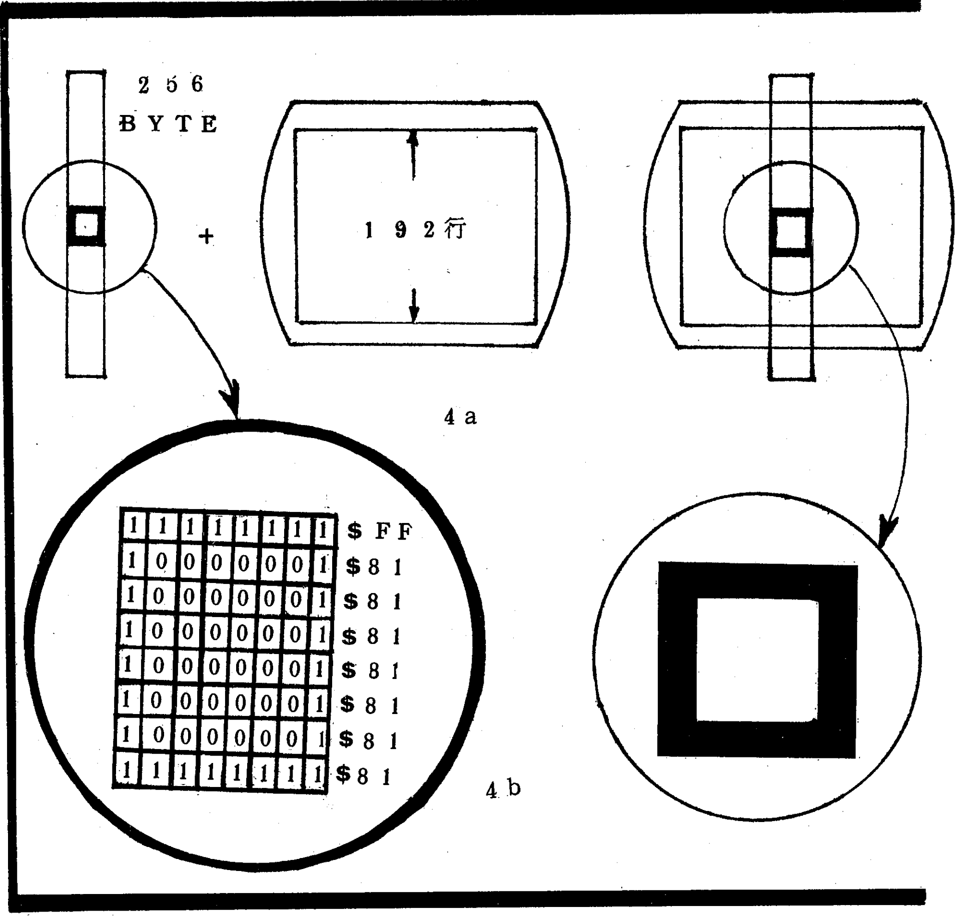
再者我們將8個BYTES $FF，$81，$81，$81，$81，$81，$81，$FF（255，129，129，129，129，129，129，255）結合在一起，那麼便會得到一個正方形的圖形（圖3）

每一個PLAYER所佔用的256個BYTE，用來代表PL－AYER的形狀，就是這個道理。因爲每個PLAYER有256

個ＢＹＴＥ，所以每個ＰＬＡＹＥＲ可以有２５６行高，而因爲ＰＬＡＹＥＲ有８行闊。這裏ＰＬＡＹＥＲ高度的每一行，是與ＧＲＡＰＨＩＣＳ｀ＭＯＤＥ　８的一點的高度相同，而ＰＬＡＹＥＲ闊度的一行，是與ＧＲＡＰＨＩＣＳ　ＭＯＤＥ　７的一點的闊度相同。

ＰＬＡＹＥＲ在畫面顯示的形式：

好了，ＰＬＡＹＥＲ的形狀知道了，ＰＬＡＹＥＲ在畫面顯示出來又是怎樣形式的呢？每個ＰＬＡＹＥＲ有２５６行高。而畫面有１９２行高，畫面出現ＰＬＡＹＥＲ的形狀就正如將ＰＬＡＹＥＲ的２５６個ＢＹＴＥ放在畫面上，如圖４ａ所示。



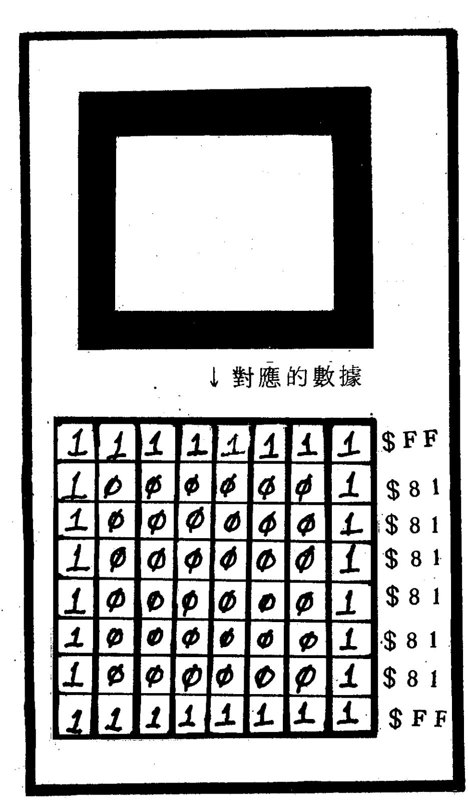


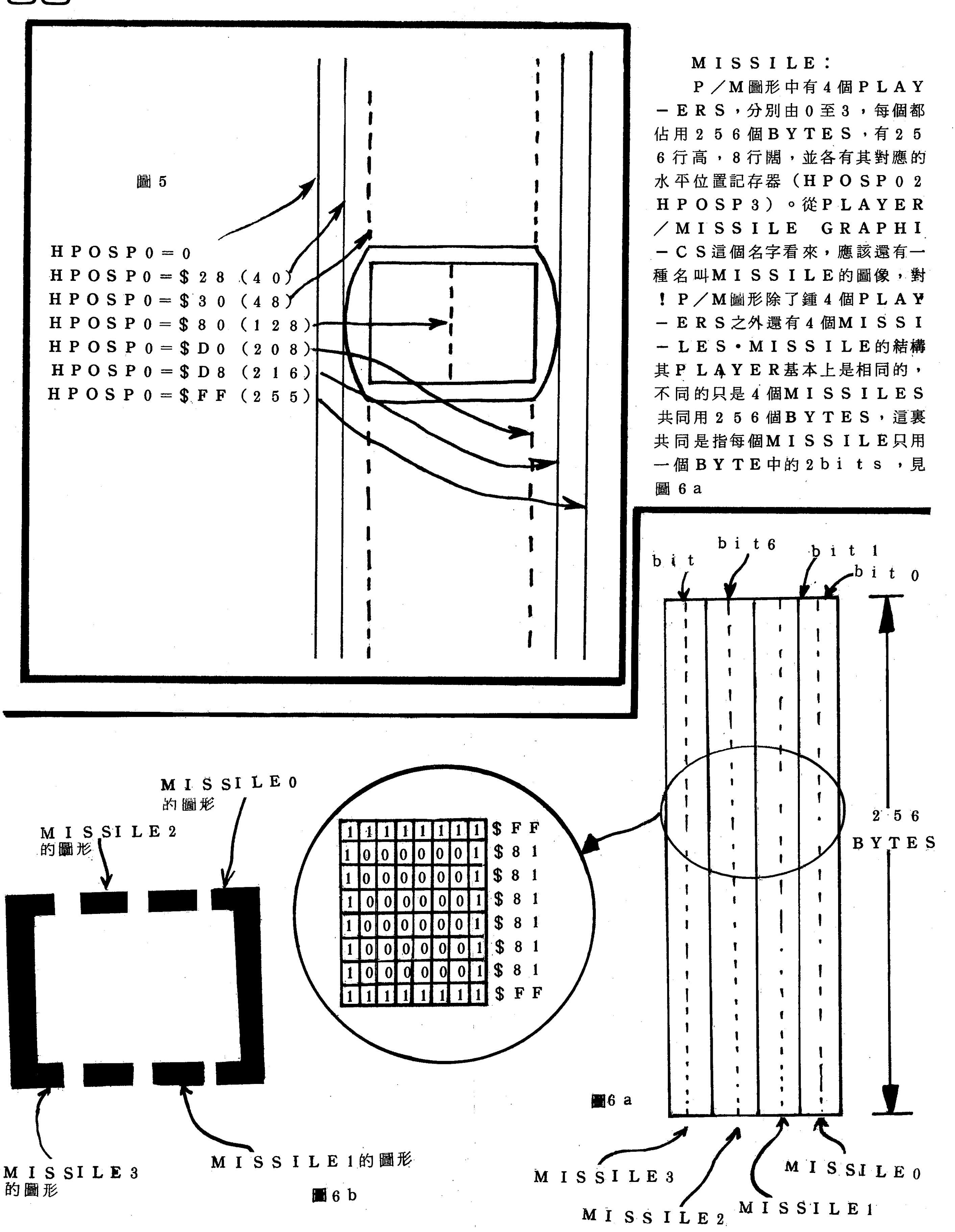
圖３

例如２５６個ＢＹＴＥ中間的８個是＄ＦＦ，＄８１，＄８１，＄８１，＄８１，＄８１，＄８１，＄ＦＦ（２５５，１２９，１２９，１２９，１２９，１２９，１２９，２５５）那麽在畫面中間便會顯示出一個正方形，如圖４ｂ所示。

ＰＬＡＹＥＲ的水平位置：

ＰＬＡＹＥＲ的垂直位置，由前段已知道是對應於ＰＬＡＹＥＲ所用的２５６個ＢＹＴＥ，那即是說，若在２５６個ＢＹＴＥ的較後部份有數據，那麽顯示出畫面的圖形便會在較低的位置，若在２５６個ＢＹＴＥＳ的較頭有數據，那麽顯示出畫面的圖形便會在較高的位置。致於ＰＬＡＹＥＲ的水平位置又如何呢？決定ＰＬＡＹＥＲ的水平位置既簡單又方便。在記憶中有一個ＲＥＧＩＳＴＥＲ名叫ＨＰＯＳＰ０，全寫是ＨＯＲＩＺＯＮＴＡＬ　ＰＯＳＩＴＩＯＮ　ＯＦ　ＰＬＡＹＥＲ０意即ＰＬＡＹＥＲ０的水平位置，只要我們將需要的水平位置記入這個ＲＥＧＩＳＴＥＲ便會將ＰＬＡＹＥＲ０移到適當的水平位置。例如將＄８０（１２８）記入去，ＰＬＡＹＥＲ０便會顯示在畫面的中央，而再將＄６０（９６）記入去便會令ＰＬＡＹＥＲ０立刻移到較左的位置。因爲ＨＰＯＳＰ０能記０至２５５的數，而畫面只有１６０行，所以將０記入ＨＰＯＳＰ０會使ＰＬＡＹＥＲ０移到看不見的最左面，同樣，若將２５５記入去便會使ＰＬＡＹＥＲ０移到看不見的最右面。到底畫面（看得見）的最左和最右面在那處呢：這會因各部不同的電視機而有所相差，大致上是＄２８（４０）（左）和＄Ｄ８（２１６）（右），不過普通ＧＲＡＰＨＩＣ　ＭＯＤＥ的最左和最右面是固定的，相等於ＨＰＯＳＰ０的＄３０（４８）（左）和＄ＤＯ（２０８）（右），見圖５。





MISSILE:

P／M圖形中有4個PLAYERS，分別由0至3，每個都佔用256個BYTES，有256行高，8行闊，並各有其對應的水平位置記存器（HPOSP0 2 HPOSP3）。從PLAYER／MISSILE GRAPHICS這個名字看來，應該還有一種名叫MISSILE的圖像，對！P／M圖形除了鍾4個PLAYERS之外還有4個MISSILES・MISSILE的結構其PLAYER基本上是相同的，不同的只是4個MISSILES共同用256個BYTES，這裏共同是指每個MISSILE只用一個BYTE中的2bits，見圖6a

圖6a

圖6b

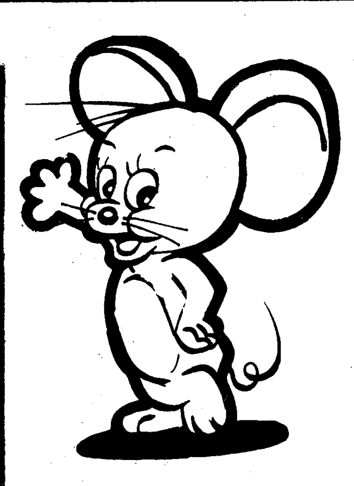
舉個例，若這256個BYTE中間8個BYTE是$FF，$81，$81，$81，$81，$81，$81，$FF（255，129，129，129，129，129，129，255）則各MISSILE在畫面顯示出來圖形便如6b一般。當然，每個MISSILE都有其相對應的水平位置記存器（HORIZONTAL POSITION REGISTER）HPOSM0～HPOSM3來記下各MISSILE的水平位置。同樣，MISSILES亦用256個BYTE來代表其形狀，所以亦和PLAYERS一樣有256行高，不過因爲MISSILES的闊度用2bits來代表，所以MISSILES只有兩行闊。

P／M所佔用的記憶區（MEMORY AREA USED IN PLAYER／MISSIILE GRAPHICS）：

前段說過共有4個PLAYERS和4個MISSILES，每個PLAYER佔用256個BYTES而4個MISSILE又用256個，總共用34×256＋256＝1280個BYTES，到底這1280個BYTES在那裏？之間又有何關係？

在記憶中有一個REGIESTER名叫PMBAS全名是PLAYER／MISSILE ADDRESS BASE，它記下3P／M圖形所用的記憶區的位置，如圖7所示4個。MISSILES所佔用的256個BYTES就在PMBASE＋$300（768）至PMBASE＋$3FF（2023）處，而PLAYER0所佔用的256個BYTE就在PMBASE＋$400（1024）至PMBASE＋$FF（1279）處。（請參看圖7）

PLAYER MISSILE GRAPHICS IS VERY EASY!!!



PMBASE→	UNUSE (FREE)				
PMBAS＋$300（768）→	M3	M2	M1	M0	總共佔用 4×256＋256＝1280（$500）BYTES
PMBAS＋$400（1024）→	PLAYER0				
PMBAS＋$500（1280）→	PLAYER1				
PMBAS＋$600（1536）→	PLAYER2				
PMBASE＋700（1792）→	PLAYER3				
PMBASE＋$800（12048）→					

因爲PMBASE是可以改變的，所以我們可以任意指定P／M圖形所佔用的記憶的地址。

如何使P／M工作（HOW TO ENABLE PLAYER／MISSILE GRAPHICS）：

在說明如何使P／M工作前，先談談一些有關P／M圖形的REGIESTERS：

DMACTL（$22F）（559）
(DIREIT MEMORY ACCESS CONTROL)

x	x	x	b_4	b_3	b_2	x	x

b_2 ＝1 ENABLE MISSICE DMA

b_3 ＝1 ENABLE PLAYER DMA

x ＝無關。

這個記存器是掩制ANTIC *用DMA取P／M之資料的，若要P／M工作，b b b 必定要是1。

*GTIA和 *ANTIC 是ATARI COMPUTER 中用以控制GRAPHICS的C－hip（晶片）

GRACTL（$DOID）
(GRAPHICS CONTROL)

	x	b_1	b_0

bo ＝1 ENABLE PLA－YER圖形
b1 ＝1 ENABLE MIS－SILE圖形

這個記存器是控制ANTIC轉送資料到GTIA *的，若要P M ／ 工作，bo b 必定要是1

PM BASE（$D407）
（ P／M AODRFSS BASE ）

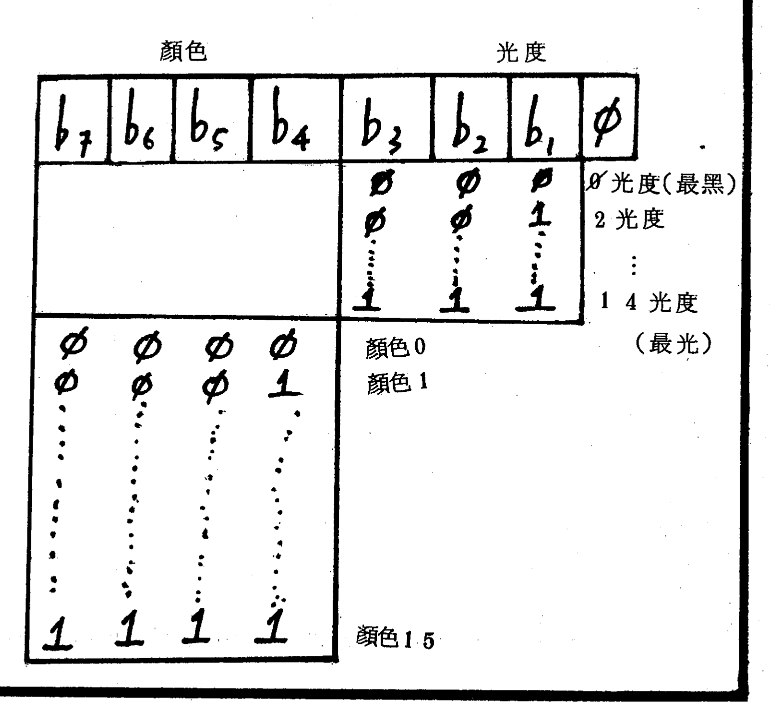
b_7	b_6	b_5	b_4	b_3	∅	∅	∅

	∅	∅

P／MBASE只有一個BY－TES，但所有地址都是兩BY－TES，所以PMBASE是視作P／M所佔用記憶地址的HIG－H BYTE，這即是說，PM－BASE所記的數不是實際的地址而實際的地址是256 *PMB－ASE。此外，因爲PHBAS－E之bo ，b 及b 是0所以PMBASE會是8的倍數而實際的P／M地址是4K的倍數。例如將$4。（64）記入PMBAS－E即代表P／M圖形所佔用的地址是$4000開始，若將$4，記入PMBASE會得到與$40同樣的結果因PMBASE的bi－t 是0。（勿忘記PMBASE開始的$300（768）個BY－TES是沒有用到的。所以實際用到的1280個BYTES是由$4300開始）。

與普通GRAPHIC MO－DE一樣，P／M圖形亦有它的顏色記存器，COLPM0是PL－AYER0和MISSILE0的顏色等等。例如一個$3A就是光度10的紅色。

PLAY／MISSILE
COLOR REGISTER
COLPM0～COLPM3（$2C0～2C3）



```
062F A900   1510          LDA  #0
0631 8D01D0 1520          STA  HPOSP1       ; OTHERS = 0
0634 8D02D0 1530          STA  HPOSP2
0637 8D03D0 1540          STA  HPOSP3
063A 8D04D0 1550          STA  HPOSM0
063D 8D05D0 1560          STA  HPOSM1
0640 8D06D0 1570          STA  HPOSM2
0643 8D07D0 1580          STA  HPOSM3
            1590 ;
0646 A93A   1600          LDA  #$3A         ; RED WITH LUM. 10
0648 8DC002 1610          STA  COLPM0
064B A900   1620          LDA  #0           ; OTHERS ARE BLACK
064D 8DC102 1630          STA  COLPM1
0650 8DC202 1640          STA  COLPM2
0653 8DC302 1650          STA  COLPM3
            1660 ;
0656 60     1670          RTS               ; RETURN WHEN DONE
            1680 ;
            1690 PLOTBL
            1700 ; PLAYER0 SHAPE TABLE
            1710 ;
0657 FF     1720          .BYTE $FF,$81,$81,$81
0658 81
0659 81
065A 81
065B 81     1730          .BYTE $81,$81,$81,$FF
065C 81
065D 81
065E FF
            1740 ;
065F        1750          .END              ; END OF ROUTINE

=022F DMACTL    =D01D GRACTL    =D407 PMBASE    =02C0 COLPM0
=02C1 COLPM1    =02C2 COLPM2    =02C3 COLPM3    =D000 HPOSP0
=D001 HPOSP1    =D002 HPOSP2    =D003 HPOSP3    =D004 HPOSM0
=D005 HPOSM1    =D006 HPOSM2    =D007 HPOSM3    =4000 PMADD
 0619 LOOP1      0621 LOOP       0657 PLOTBL
```

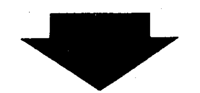
文章一開始便說P／M圖形有良好移動的特性，到底怎樣供P／M圖形移動呢？現在讓我說明：

垂直移動：因爲P／M圖形在畫面上是一度空間顯示的，所以要P／M圖形垂直移動，其實只是將P／M所用的記憶位置前後移動，非常簡單。

水平移動：不用說明了，P／M圖形用HP S記存器來控制它的水平位置，要P／M圖形在水平移動，只要改變HPOS記存器便可。

因P／M圖形的水平和垂直移動是獨立無關的，所以要P／M圖形作出各種不同方向的移動只要用不同的水平和垂直移動組合便可完成。

以下是兩段（BASIC和6502ASSEMBLY）程式使P／M作垂直和水平移動：



PROGRAM 3

```
1000 REM * ROUTINES USED TO MOVE PLAYER 0
1010 REM *
1020 REM *                BY  BAN SZETO
1030 REM *
1040 HPOSP0=13*4096:REM $D000
1050 PMADD=4*4096:REM $4000
1060 REM *
1070 REM * MOVE RIGHT
1080 HPOSU=HPOSU+1
1090 POKE HPOSP0,HPOSU
1100 RETURN
1110 REM *
1120 REM * MOVE LEFT
1130 HPOSU=HPOSU-1
1140 POKE HPOSP0,HPOSU
1150 RETURN
1160 REM *
1170 REM * MOVE UP
1180 VPOSU=VPOSU-1
1190 USADD=PMADD+4*256+VPOSU
1200 FOR N=0 TO 8
1210 POKE USADD+N,PEEK(USADD+N+1)
1220 NEXT N
1230 RETURN
1240 REM *
1250 REM * MOVE DOWN
1260 USADD=PMADD+4*256+VPOSU
1270 FOR N=8 TO 0 STEP -1
1280 POKE USADD+N,PEEK(USADD+N-1)
1290 NEXT N
1300 VPOSU=VPOSU+1
1310 RETURN
```

PROGRAM 4

```
0000          1000        *=    $600
              1010 ; ROUTINE MOVE PLAYER0 LEFT AND RIGHT
              1020 ;
              1030 ;                    BY  BAN SZETO
              1040 ;
4000          1050 PMADD  =     $4000
D000          1060 HPOSP0 =     $D000
0600          1070 HPOSU  *=    *+1
0601          1080 VPOSU  *=    *+1
              1090 ;
              1100 MVLEFT
0602 CE0006   1110        DEC   HPOSU
0605 AD0006   1120        LDA   HPOSU
0608 8D00D0   1130        STA   HPOSP0
060B 60       1140        RTS
```

```
               1150 ;
               1160 MVRGHT
060C EE0006    1170          INC   HPOSU
060F AD0006    1180          LDA   HPOSU
0612 8D00D0    1190          STA   HPOSPO
0615 60        1200          RTS
               1210 ;
               1220 ;
               1230 ; ROUTINE MOVE PLAYERO UP AND DOWN
               1240 ;
               1250 MVUP
0616 A200      1260          LDX   #0
0618 AC0106    1270          LDY   VPOSU
061B B90044    1280 LOOP     LDA   PMADD+$400,Y
061E 99FF43    1290          STA   PMADD+$400-1,Y
0621 C8        1300          INY
0622 E8        1310          INX
0623 E009      1320          CPX   #9
0625 D0F4      1330          BNE   LOOP
               1340 ;
0627 CE0106    1350          DEC   VPOSU
062A 60        1360          RTS
               1370 ;
               1380 ;
               1390 MVDOWN
062B A208      1400          LDX   #8
062D AD0106    1410          LDA   VPOSU
0630 18        1420          CLC
0631 6908      1430          ADC   #8
0633 A8        1440          TAY
0634 B9FF43    1450 LOOP2    LDA   PMADD+$400-1,Y
0637 990044    1460          STA   PMADD+$400,Y
063A 88        1470          DEY
063B CA        1480          DEX
063C 10F6      1490          BPL   LOOP2
               1500 ;
063E EE0106    1510          INC   VPOSU
0641 60        1520          RTS
               1530 ;
               1540 ;
0642           1550          .END

=4000 PMADD      =D000 HPOSPO     0600 HPOSU     0601 VPOSU
 0602 MVLEFT      060C MVRGHT     0616 MVUP      061B LOOP
 062B MVDOWN      0634 LOOP2
```

以下提供3二個完整的（BA－SIC和6502 ASSEM－BLY）示範程式作爲讀者參考的。

PROGRAM 5

```
10 REM * PLAYER/MISSILE GRAPHICS DEMONSTRATION PROGRAM
20 REM *
30 REM *                    BY  BAN SZETO
40 REM *
100 GOSUB 500:REM ENABLE PLAYER/MISSILE
110 IF STICK(0)=7 AND HPOSU<255 THEN GOSUB 1080
120 IF STICK(0)=11 AND HPOSU>0 THEN GOSUB 1130
130 IF STICK(0)=14 AND VPOSU>0 THEN GOSUB 1180
140 IF STICK(0)=13 AND VPOSU<255 THEN GOSUB 1260
150 GOTO 110
500 REM * ROUTINE USED TO ENABLE PLAYER/MISSILE GRAPHICS
510 REM *
540 DMACTL=559:REM DIRECT MEMORY ACCESS CONTROL
550 GRACTL=53277:REM GRAPHICS CONTROL
560 PMBASE=54279:REM PLAYER/MISSILE ADDRESS BASE
570 COLPM0=704:REM PLAYER/MISSILE COLOR
580 HPOSP0=53248:REM HORIZONTAL POSITION OF PLAYER 0
590 HPOSM0=53252:REM HORIZONTAL POSITION OF MISSILE 0
600 PMADD=16384:REM P/M MEMORY USED START ADDRESS
610 REM *
620 POKE DMACTL,62
630 POKE GRACTL,3
640 POKE PMBASE,PMADD/256
650 REM *
660 USADD=PMADD+4*256
670 FOR N=0 TO 255
680 POKE USADD+N,0
690 NEXT N
700 REM *
710 USADD=PMADD+4*256+128
720 FOR N=0 TO 7
730 READ SHAP
740 POKE USADD+N,SHAP
750 NEXT N
755 VPOSU=128
760 REM *
770 FOR N=1 TO 3
780 POKE COLPM0+N,0
790 NEXT N
800 POKE COLPM0,3*16+10
810 REM *
820 FOR N=1 TO 7
830 POKE HPOSP0+N,0
840 NEXT N
850 POKE HPOSP0,128
855 HPOSU=128
860 REM *
870 REM PLAYER 0 SHAP TABLE
880 DATA 255,129,129,129
890 DATA 129,129,129,255
900 RETURN
1000 REM * ROUTINES USED TO MOVE PLAYER 0
1010 REM *
1040 HPOSP0=13*4096:REM $D000
1050 PMADD=4*4096:REM $4000
```

```
1060 REM *
1070 REM * MOVE RIGHT
1080 HPOSU=HPOSU+1
1090 POKE HPOSPO,HPOSU
1100 RETURN
1110 REM *
1120 REM * MOVE LEFT
1130 HPOSU=HPOSU-1
1140 POKE HPOSPO,HPOSU
1150 RETURN
1160 REM *
1170 REM * MOVE UP
1180 VPOSU=VPOSU-1
1190 USADD=PMADD+4*256+VPOSU
1200 FOR N=0 TO 8
1210 POKE USADD+N,PEEK(USADD+N+1)
1220 NEXT N
1230 RETURN
1240 REM *
1250 REM * MOVE DOWN
1260 USADD=PMADD+4*256+VPOSU
1270 FOR N=8 TO 0 STEP -1
1280 POKE USADD+N,PEEK(USADD+N-1)
1290 NEXT N
1300 VPOSU=VPOSU+1
1310 RETURN
```

PROGRAM 6

```
                1000 ; PLAYER/MISSILE GRAPHICS DEMONSTRATION PROGRAM
                1010 ;
                1020 ;                    BY  BAN SZETO
                1030 ;
0000            1040          *=     $600
022F            1050 DMACTL  =     $22F       ; DIRECT MEMORY ACCESS CONTROL
D01D            1060 GRACTL  =     $D01D      ; GRAPHICS CONTROL
D407            1070 PMBASE  =     $D407      ; PLAYER/MISSILE ADDRESS BASE
02C0            1080 COLPM0  =     $2C0       ; PLAYER/MISSILE0 COLOR
02C1            1090 COLPM1  =     $2C1       ;                 1
02C2            1100 COLPM2  =     $2C2       ;                 2
02C3            1110 COLPM3  =     $2C3       ;                 3
D000            1120 HPOSP0  =     $D000      ; HORIZONTAL POSITION P0
D001            1130 HPOSP1  =     $D001      ;                     P1
D002            1140 HPOSP2  =     $D002      ;                     P2
D003            1150 HPOSP3  =     $D003      ;                     P3
D004            1160 HPOSM0  =     $D004      ;                     M3
D005            1170 HPOSM1  =     $D005      ;                     M1
D006            1180 HPOSM2  =     $D006      ;                     M2
D007            1190 HPOSM3  =     $D007      ;                     M3
4000            1200 PMADD   =     $4000      ; MEMORY USED IN PLAYER/MISSILE
                1210 ;
0278            1220 STICK0  =     $278       ; JOYSTICK #0
000E            1230 UP      =     14         ; JOYSTICK UP
```

```
000D          1240 DOWN    =     13          ; JOYSTICK DOWN
000B          1250 LEFT    =     11          ; JOYSTICK LEFT
0007          1260 RIGHT   =     7           ; JOYSTICK RIGHT
              1270 ;
              1280 MAIN
0600 202F06   1290         JSR   ENPM        ; ENABLE PLAYER MISSILE GRAPHICS
              1300 ;
0603 AD7802   1310 AGAIN   LDA   STICK0
0606 C90E     1320         CMP   #UP         ; IS IT UP?
0608 D003     1330         BNE   NEXT1       ; NO, TEST DOWN
060A 20AC06   1340         JSR   MVUP        ; YES, MOVE UP
              1350 ;
060D C90D     1360 NEXT1   CMP   #DOWN       ;IS IT DOWN?
060F D003     1370         BNE   NEXT2       ; NO, TEST LEFT
0611 20C106   1380         JSR   MVDOWN      ; YES, MOVE DOWN
              1390 ;
0614 C90B     1400 NEXT2   CMP   #LEFT       ;IS IT LEFT?
0616 D003     1410         BNE   NEXT3       ; NO, TEST RIGHT
0618 209806   1420         JSR   MVLEFT      ; YES, MOVE LEFT
              1430 ;
061B C907     1440 NEXT3   CMP   #RIGHT      ;IS IT RIGHT?
061D D003     1450         BNE   NEXT4       ; NO, DELAY AND LOOP AGAIN
061F 20A206   1460         JSR   MVRGHT      ; YES, MOVE RIGHT
              1470 ;
              1480 NEXT4
0622 A200     1490         LDX   #0
0624 A010     1500         LDY   #$10
              1510 DELAY
0626 CA       1520         DEX
0627 D0FD     1530         BNE   DELAY
0629 88       1540         DEY
062A D0FA     1550         BNE   DELAY
              1560 ;
062C 4C0306   1570         JMP   AGAIN       ; AGAIN AND AGAIN
              1580 ;
              1590 ;
              1600 ; * ROUTINE USED TO ENABLE PLAYER/MISSILE
              1610 ;
              1620 ENPM
062F AD2F02   1630         LDA   DMACTL
0632 091C     1640         ORA   #$1C        ; MAKE BIT3 AND BIT4 OF DMACTL 1
0634 8D2F02   1650         STA   DMACTL
              1660 ;
0637 AD1DD0   1670         LDA   GRACTL
063A 0903     1680         ORA   #$03        ; MAKE BIT0 AND BIT1 OF GRACTL 1
063C 8D1DD0   1690         STA   GRACTL
              1700 ;
063F A940     1710         LDA   #PMADD/256 ; STORE MEMORY USED ADDRESS IN
                                               PMBASE
0641 8D07D4   1720         STA   PMBASE
              1730 ;
              1740 ;
```

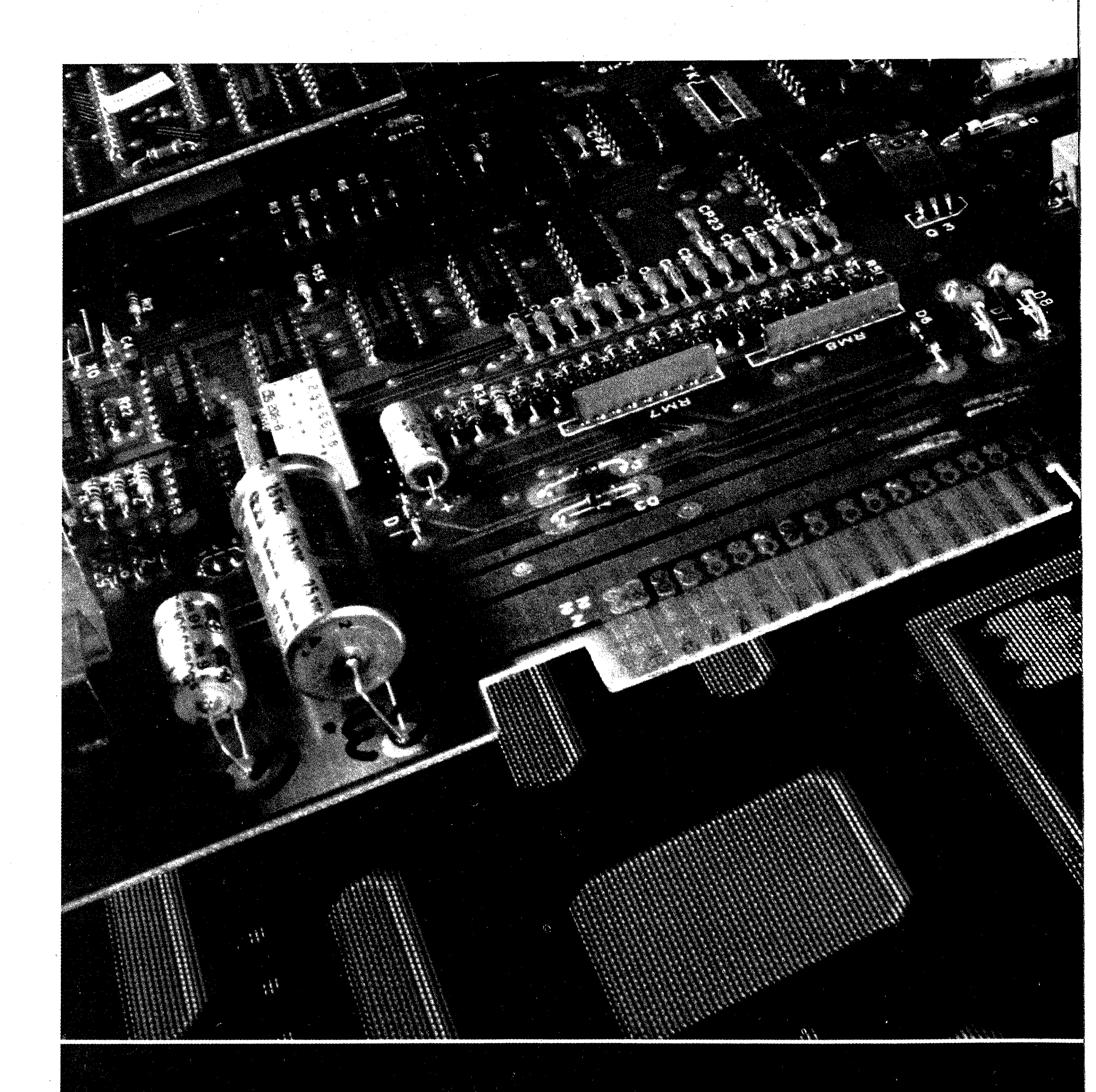
```
0644 A200    1750          LDX  #0          ; CLEAR MEMORY USED IN P/M
0646 A900    1760          LDA  #0
0648 9D0044  1770 LOOP1    STA  PMADD+$400,X
064B CA      1780          DEX
064C D0FA    1790          BNE  LOOP1
             1800 ;
064E A207    1810          LDX  #7
0650 BD8E06  1820 LOOP     LDA  PLOTBL,X    ; STA PLAYER'S SHAP IN P/M MEMORY
0653 9D8044  1830          STA  PMADD+$400+$80,X
0656 CA      1840          DEX
0657 10F7    1850          BPL  LOOP
0659 A980    1860          LDA  #$80
065B 8D9706  1870          STA  VPOSU
             1880 ;
065E A980    1890          LDA  #$80
0660 8D00D0  1900          STA  HPOSP0      ; HOR. POS. OF PLAYER0 = 128
0663 8D9606  1910          STA  HPOSU
0666 A900    1920          LDA  #0
0668 8D01D0  1930          STA  HPOSP1      ; OTHERS = 0
066B 8D02D0  1940          STA  HPOSP2
066E 8D03D0  1950          STA  HPOSP3
0671 8D04D0  1960          STA  HPOSM0
0674 8D05D0  1970          STA  HPOSM1
0677 8D06D0  1980          STA  HPOSM2
067A 8D07D0  1990          STA  HPOSM3
             2000 ;
067D A93A    2010          LDA  #$3A        ; RED WITH LUM. 10
067F 8DC002  2020          STA  COLPM0

0682 A900    2030          LDA  #0          ; OTHERS ARE BLACK
0684 8DC102  2040          STA  COLPM1
0687 8DC202  2050          STA  COLPM2
068A 8DC302  2060          STA  COLPM3
             2070 ;
068D 60      2080          RTS              ; RETURN WHEN DONE
             2090 ;
             2100 PLOTBL
             2110 ; PLAYER0 SHAPE TABLE
             2120 ;
068E FF      2130          .BYTE $FF,$81,$81,$81
068F 81
0690 81
0691 81
0692 81      2140          .BYTE $81,$81,$81,$FF
0693 81
0694 81
0695 FF
             2150 ;
0696         2160 HPOSU    *=   *+1         ; HORIZONTAL POSITION USED
0697         2170 VPOSU    *=   *+1         ; VERTICAL POSITION USED
             2180 ;
             2190 MVLEFT
0698 CE9606  2200          DEC  HPOSU       ; MOVE PLAYER LEFT
069B AD9606  2210          LDA  HPOSU
```



```
069E 8D00D0 2220        STA   HPOSP0
06A1 60     2230        RTS
            2240 ;
            2250 MVRGHT
06A2 EE9606 2260        INC   HPOSU        ; MOVE PLAYER RIGHT
06A5 AD9606 2270        LDA   HPOSU
06A8 8D00D0 2280        STA   HPOSP0
06AB 60     2290        RTS
            2300 ;
            2310 ;
            2320 MVUP
06AC A200   2330        LDX   #0           ; MOVE PLAYER UP
06AE AC9706 2340        LDY   VPOSU
06B1 B90044 2350 LOOP3  LDA   PMADD+$400,Y
06B4 99FF43 2360        STA   PMADD+$400-1,Y
06B7 C8     2370        INY
06B8 E8     2380        INX
06B9 E009   2390        CPX   #9
06BB D0F4   2400        BNE   LOOP3
            2410 ;
06BD CE9706 2420        DEC   VPOSU
06C0 60     2430        RTS
            2440 ;
            2450 ;
            2460 MVDOWN
06C1 A208   2470        LDX   #8           ; MOVE PLAYER DOWN
06C3 AD9706 2480        LDA   VPOSU
06C6 18     2490        CLC
06C7 6908   2500        ADC   #8
06C9 A8     2510        TAY
06CA B9FF43 2520 LOOP2  LDA   PMADD+$400-1,Y
06CD 990044 2530        STA   PMADD+$400,Y
06D0 88     2540        DEY
06D1 CA     2550        DEX
06D2 10F6   2560        BPL   LOOP2
            2570 ;
06D4 EE9706 2580        INC   VPOSU
06D7 60     2590        RTS
            2600 ;
            2610 ;
06D8        2620        .END

=022F DMACTL     =D01D GRACTL     =D407 PMBASE     =02C0 COLPM0
=02C1 COLPM1     =02C2 COLPM2     =02C3 COLPM3     =D000 HPOSP0
=D001 HPOSP1     =D002 HPOSP2     =D003 HPOSP3     =D004 HPOSM0
=D005 HPOSM1     =D006 HPOSM2     =D007 HPOSM3     =4000 PMADD
=0278 STICK0     =000E UP         =000D DOWN       =000B LEFT
=0007 RIGHT       0600 MAIN        062F ENPM        0603 AGAIN
 060D NEXT1       06AC MVUP        0614 NEXT2       06C1 MVDOWN
 061B NEXT3       0698 MVLEFT      0622 NEXT4       06A2 MVRGHT
 0626 DELAY       0648 LOOP1       0650 LOOP        068E PLOTBL
 0697 VPOSU       0696 HPOSU       06B1 LOOP3       06CA LOOP2
```



Inside Pac-Man

6502
80

食鬼遊戲（Pac－Man）現已成爲國際高級藝術品，兩年內三十多萬部食鬼遊戲機已打入了世界各類的商場、商店、會所及酒店。狼吞虎嚥了大量的美金、北歐幣及日圓。

在今年前些時所推出的家庭式食鬼遊戲，其受歡迎程式比任何受歡迎的電影更爲熱烈。實際上，家庭式Pac－Man遊戲帶給了華納廣播公司（Warner Co－mmunication Longlomerate）一筆很大的利潤，是它公司本身的電影及錄音製作所不能賺到的。Pac－Man是第一種能吸引大量女性的電子遊戲機。根據佛洛伊德學說的解釋，Pac－Man的面貌有如一個微笑的鈕扣，它會吞噬它的獵物，而不是射殺他們，使婦女們玩得更覺趣味。

機械、電子和電視的系統使遊戲者能控制Pac－Man的行動，正如以前由微型電腦所控制的一樣，只是安裝多了數塊集成電路矽片。Masaya Nakamu－ra，和路廸士尼電視遊戲公司及Namco的設計組共同製造了Pac－Man和四個顏色鮮艷的鬼魅外形，在遊戲進行時，他們會一直沿着發磷光的路線對Pac－Man尾追不捨，兼且會發出電子聲響效果及音樂。

到底Pac－Man電子遊戲機中有甚麽秘密？能使彩色電視的螢光幕變爲一種受愛好的冒險呢？

或者你不會奇怪，在電子遊戲機中，有一塊主要邏輯電路板，在它的上面裝置有無數的電路矽片，近似於私人微型電腦的電子零件部份。

實際上，開動電子遊戲的Z－80微型處理，是同樣用於很多小型商業之微型電腦系統，以作爲中央資料處理器。不像一般普通的多用途電腦，只要輸入程序（Pro－gram）便可處理不同類別及性質的工作。一部電子遊戲機的電



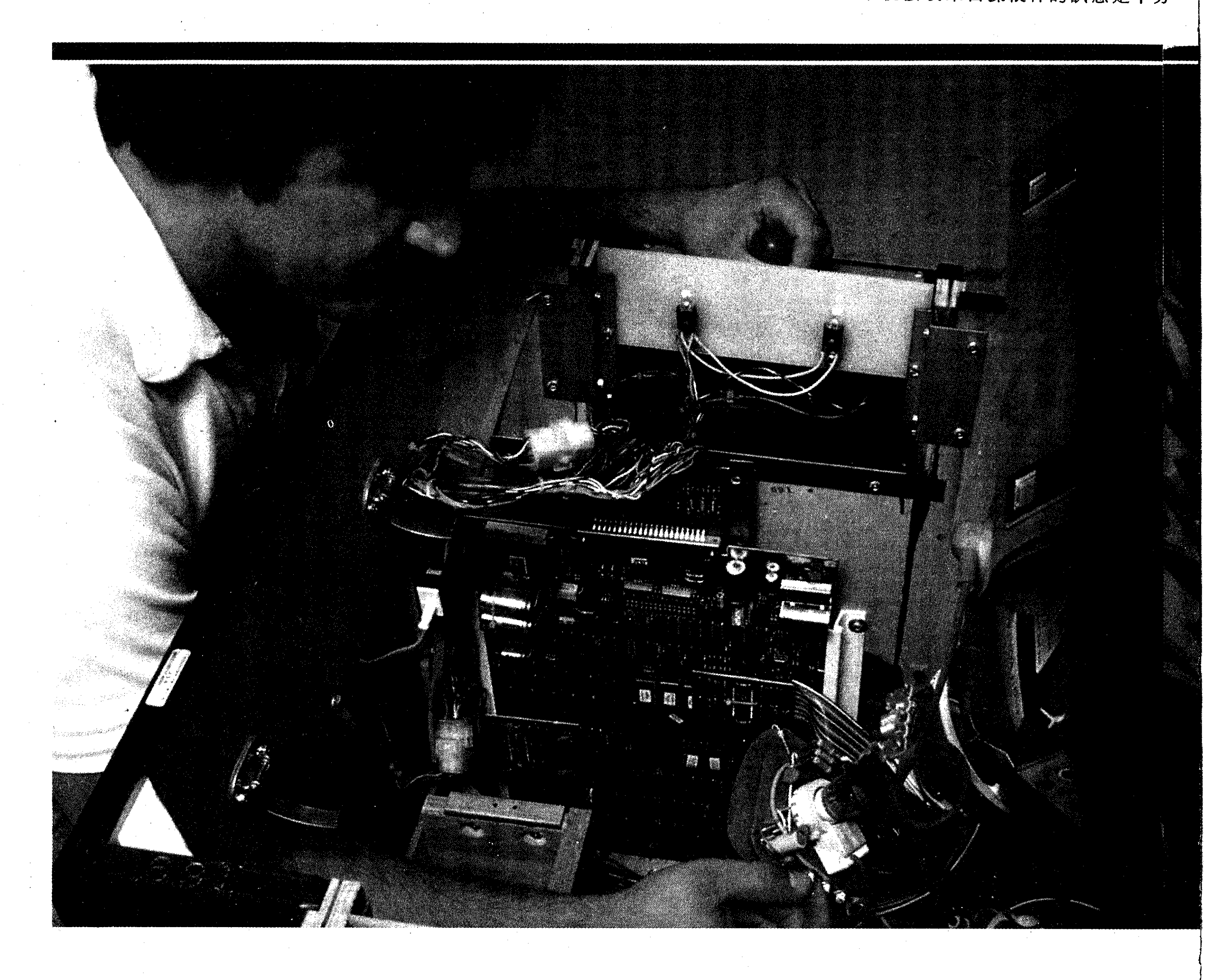
腦性能是被「局限」了的，它最重要的責任是作爲人類及電視螢幕的連繫者。它不需要像微型電腦般，要處理來自鍵盤的資料，人類的語言，大量資料貯藏的設計如磁碟機及一部相連的速印機等。一架電子遊戲機能利用它的集成電路（Pac－Man 有八十四塊電路在主要邏輯板上）去造成複雜的電視影像，生動的形像及複雜的音響效果。

一家美國公司（Midway Division of Bally）已製成及註冊了一種有水準的美國Pac－Man電子遊戲機，可以接收從十五處不同來源所發出的資料，比如入幣感應器、選擇單雙人遊戲的按鈕，及最重要的是一副機器控制了四個開關，指示出那操縱桿是移向那一方位。

於任何電腦設計中，一副石英鐘電路爲每一種功能提供計時服務。電子遊戲機的中央資料處理之操作速度比同類型的蘋果牌微型電腦（Apple Ⅱ Microcomputer）的計時器速度快三倍。因此，遊戲機能發出穩定及準確的反應。

電子遊戲機螢幕上的影像是由兩個集成電路矽片名叫只讀記憶系統（ROMS）。這些只讀記憶矽片是貯藏程序指示的地方，它規定了螢幕上的影像範圍及其相互位置。每一塊矽片能夠貯藏4096個單字。每一塊矽片提供指示，使遊戲機螢幕上顯示出應走的路線及路線中之障礙物。其它的矽片貯藏了詳細的資料，使這些障礙物能重畫Pac－Man及其追踪者。任何螢幕上影像的輪廓，我們都能解釋爲簡單的xy座標的相互關係。一個影像的顏色是決定於其它的數字程序。更且，電腦使用了四個不同寬度的口形，使之迅速交替變換，造成Pac－Man的大口能開能合。

要使一個影像能沿着它的路線走，訊號會增加影像的x或y軸座標。當遊戲者推着Pac－Man操縱桿時，無論觸動任何開關，都會使程序轉換指示，使x座標（負責左右的活動），或y座標（負責上下的活動）或加或減，能使整個系統接收來自操縱桿的訊息是千分



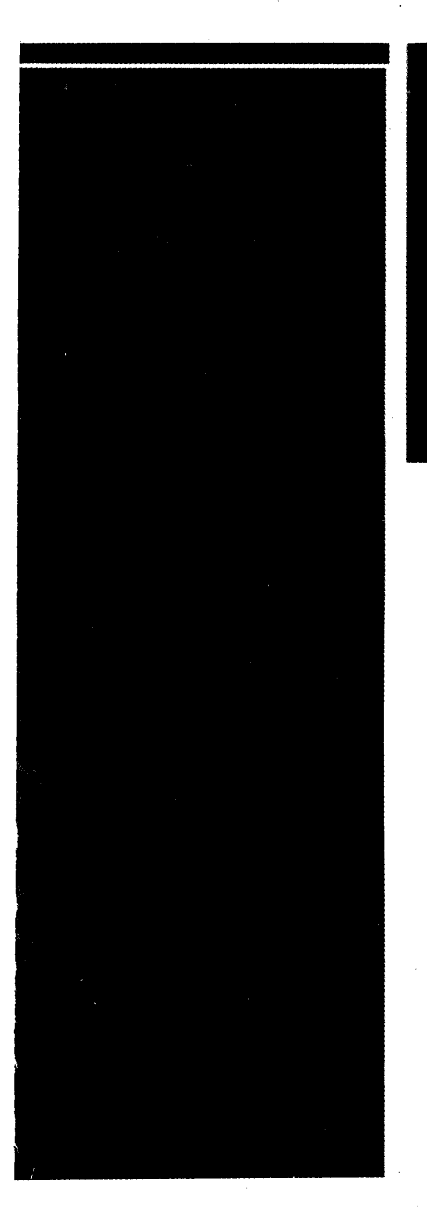
之一秒，以二進位數的零與一爲代表。電子遊戲機的資料用八個位的二進位數去處理的，即是一個byte。

當每一個byte的資料最初輸送至主要邏輯板時，它是暫時被貯藏於一塊名叫援衝矽片內（Buffer Chip），而其它所有的電腦活動則仍照常進行。當系統能接收一個byte時，它是被輸入一個名叫讀寫記憶系統的矽片內，簡稱（RAM）。六個RAM矽片能貯藏爲數共24576個bytes的資料。由於這是暫時的貯藏，所以RAM可被視爲一種拍紙薄式（Scratch Pad）記憶系統。由於RAM矽片會因電源的切斷或回按指示（RESET）失去功能，所以程序限定的遊戲規則和路線都是藏於只讀記憶系統（ROM）矽片內。當遊戲進行時，資料處理器會把程序輸送入RAM系統，使它能夠跟隨遊戲者的意願而轉變。

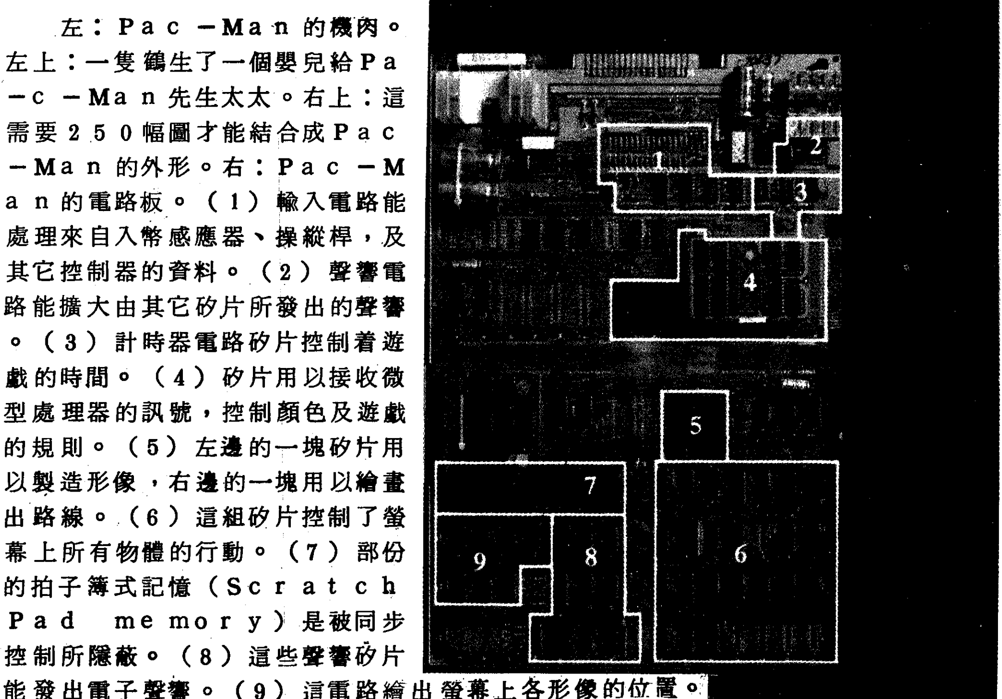
在Pac－Man程序中，操縱桿傳送每個byte的資料會翻譯爲指，利用間斷性加減xy軸的座標，以改變Pac－Man在螢幕上的位置。數據時常檢查著座標位置，規定螢幕上的迷宮路線及決定某一位置路線的交點（准許轉換方向）或那位置是被亮點所佔據（Pac－Man的速度會減慢，正當它在吃那些亮點以增加分數時）或一個大的能量點（給予Pac－Man數秒的機會去追吃其他的鬼怪。

其它Pac－Man遊戲程序的部份，或稱爲Subroutines，是規定了那些追趕Pac－Man的怪物之特別活動和特點。附加的程序指示使它們的眼睛的轉換方向及它們，身體輪廓活動。從遊戲開始的最先四秒內，那些怪物都很客氣，只是在螢幕的籠中上下走動。但過了一會兒，它們就猶如脫兎，每一怪物都跟隨Pac－Man的方位而改變它本身的行動。

那紅色的怪物，名叫影子（Shadow），它會如影隨形地追踪着Pac－Man，令它無法擺脫。至於粉紅色的怪物，名叫速度（Speedy），它的速度極快







左：Pac－Man的機肉。左上：一隻鶴生了一個嬰兒給Pac－Man先生太太。右上：這需要250幅圖才能結合成Pac－Man的外形。右：Pac－Man的電路板。（1）輸入電路能處理來自入幣感應器、操縱桿，及其它控制器的資料。（2）聲響電路能擴大由其它矽片所發出的聲響。（3）計時器電路矽片控制着遊戲的時間。（4）矽片用以接收微型處理器的訊號，控制顏色及遊戲的規則。（5）左邊的一塊矽片用以製造形像，右邊的一塊用以繪畫出路線。（6）這組矽片控制了螢幕上所有物體的行動。（7）部份的拍子薄式記憶（Scratch Pad memory）是被同步控制所隱蔽。（8）這些聲響矽片能發出電子聲響。（9）這電路繪出螢幕上各形像的位置。

GET　　　Bazo's Breaker

上圖：兩個使Pac－Man避開怪物的策略。GET比其它形式的策略優越，是在於它不須要遊戲者突然將Pac－Man轉身或吃掉那些能量點及附近小點，因爲這些行動終會使Pac－Man犯錯。當整個程式完成後，Pac－Man仍會有時間去吃掉那些點和怪物。Bazo’S Breaker很快，可在五十一秒內完成一局遊戲，遊戲者能於較短時間內贏得較高的分數。然而，兩者皆可讓遊戲者吃去所有的果子及獎分項目。

Pac-Maniacs

Pac－Man狂熱者 Pac－Man Maniacs

數學畢業生Ed Bazo說：「一個好的Pac－Man策略是在於它具有穩定性。」他和他的兩位朋友George Huang及Tom Fertad。在三文治商店裏花了無數小時研究出一個名爲GET的策略。如果謹慎地遵從這程式，可使Pac－Man像跳芭蕾舞般穿插於怪物中，而却不會被擒。因此，穩定是必需的，即使稍停於一角，也可能破壞了程式的準確性。

Ed Bazo自己則創了Bazo’s Breaker策略。它不像GET策略，來對付原先版本的電子遊戲機。Bazo’s Breaker保證遊戲者於的遊戲機版本中可玩上無限局。Bazo因用了這程式而玩了數小時之久，贏得了2,000,000分，直至他筋疲力盡始自行離去。



Fertado說：「Ed Bazo的風格是保守的，而George的風格則是侵略的，George喜行險著贏取高分。」直至目前他們仍很喜愛Pac－Man遊戲。Ed Bazo說：「我與Pac－Man遊戲實在是難捨難分，我不覺得現今有其它遊戲機可以取代Pac－Man的地位。」

，但如果Pac－Man能迅速入暗角則可以避開它。藍色的怪物名叫害羞（Bashful），它的速度慢兼且會自動離去，如果Pa－c－Man能突然轉變方向。橙色的怪物名叫Pokey，它被輸入了特有的程序，能作出一連串不同，且難以預測的行動。

雖然微型處理器能模倣隨意活動，但一直未爲採用於程序中。結果，很多Pac－Man的愛好者因瞭解了怪物活動的方式而能自創程式，使Pac－Man能避開那些怪物的追趕竟達數小時之久。

爲了要使遊戲的程式更加複雜及爲機主帶來更多的收入。在最新的遊戲機中，已替那些怪物輸入了隨意活動的程序，Pac－Man小姐（Ms. Pac－Man）。這個版本的特色是當遊戲在不斷的進行時，螢幕會產生四不同的迷宮路線圖，以代替先前的一幅通用圖。當遊戲者玩得好時，新產品能使遊戲之速度增加。螢幕上便會出現不同的水果。

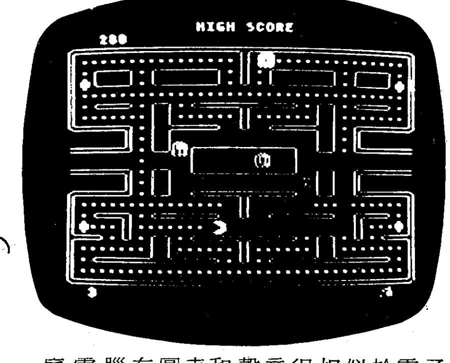
當螢幕上所有的小點爲飢餓的Pac－Man所吞噬後，遊戲的程序指示再造一個時間比前短一半的遊戲。如果在遊戲角逐的過程中，Pac－Man，被怪物所吃掉，微處理器認出座標已被擒，它會重新安排所有的變數，包括清除螢光幕上的怪物，顯示Pac－Ma－n被捉後漸漸地清失。另外一個Subroutine監察着遊戲者有沒有剩餘的Pac－Man。如果有，則遊戲再開始。如果沒有剩餘，遊戲完畢的訊號便會閃亮。另一個Subroutine則會自動地顯示出遊戲的玩耍方式。

當Pac－Man電子遊戲機開動時，便會奏出一首特別的主題音樂。這首主題音樂及Wocka－Wocka－Wocka的聲音，是發自一部藏於機中的喇叭。當Pac－Man被怪物捉到時，它會發出死亡之哀音。所有必需的資料都是以數字形式貯藏於兩塊RO－M矽片上。另外一塊矽片則作爲聲音擴音器，其它六塊則用來調整位數對聲音的轉換。

更複雜的電子遊戲機已在試驗階段中，其特點將包括無棒控制器，能靠手和眼的活動作爲輸入，詳細的圖表可與生動的電視媲美，且具有聲音的辨認及組合能力，可使遊戲者與遊戲機交談。還可以利用硬件及軟件，製造戰略，勝於只用眼手合作的方法去玩遊戲。無論這些新遊戲及其繼承產品的能力是怎樣，它們都需要相當的時間以代替Pac－Man。（完）

接 36 頁





電腦食鬼遊戲

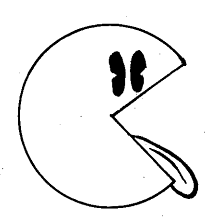
Atari家庭電腦及5200高級電視遊戲系統，有另一款食鬼遊戲。在這款遊戲中，每一樣東西都全部重新安排。這包括了不同顏色 的表現；遊戲技巧水平符號則置於螢幕右下邊。上邊積分表有"1up" "2up"及"最高分數HIGHEST SCORE"顯示。得分方法與原著版本一樣；包括在10,000分後，得到一額外食鬼怪等。此外，這款高級產品，有256點，比原來大型遊戲中的240點，多出16點。

發音音韻與原著版本略有分別。指示說明非常簡單，包括一個列表，列出了Pac－Man，Bl－inky及其它鬼怪在每一局的速度。

根據電子遊戲雜誌的主編Bill Kunkel說：「雅達利400及800形的家庭電腦有圖表和聲音很相似於電子遊戲機。」雅達利電腦的秘密？特別的共同微型處理器製造電視影像及固定的四聲道設備，包括有不同顏色的鬼，它的眼睛會跟隨着它所轉變的方向而轉動，穩定的活動，高解決困難的圖表，和熟悉的Pa－c－Man音樂及咀嚼聲音。

當按下掣後，遊戲者可選擇8種遊戲技巧速度的其中一級——由第一迷宮至第十三迷宮。

這產品的優點，在於在分解度低的家庭電視中，能有優良的圖像分析。



6502
80

PAC-MAN 縱影

在過去，我們的〝偶像〞通常是電影中的主角，卡通片中的卡通人物或甚至是喜劇中的丑角。他們成爲人們偶像的原因，大多是這些人物具有獨特的性格，他們的行爲往往使觀衆們留下深刻的印象。由於此等偶像受人擁戴，因此商人亦把握機會，在很多商品中；如Ｔ恤、咖啡杯、汽車標貼及床單，都加進了偶像的容貌，使產品變有紀念價值；從而增廣銷路。無可否認，自從此類形式商品推出後，銷量非常可觀，間接給版權擁有者帶進大量的收入。

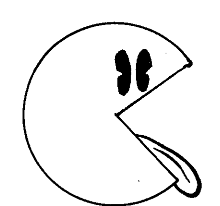
但是，一在最近所出現的一個普羅大衆偶像，不再是電影中的主角，而是一個電視遊戲中的〝生物〞。這個電視遊戲；就是食鬼〝（Ｐａｃ－Ｍａｎ）〞。其實，這個遊戲的熱潮，是始於１９８１。到１９８２，熱潮仍然未息。目前，很多地區都受到〝食鬼〞遊戲的影響。在部份電視遊戲製造商來說，他們都希望遊戲能改變成一個能適合在家庭中玩的版本。

要研究原來的〝食鬼〞遊戲版本的圖像，動作及聲音，轉變爲家庭式的版本後的質素會達到何種水準，筆者檢視了兩個手提／座枱式的〝食鬼〞遊戲機，雅達利的ＶＣ－Ｓ’４φφ／８φφ家庭、和新的５２００電視遊戲機裏的盒帶。

座枱式遊戲機

要將複雜的螢幕街道式圖像減化爲固定、眞空的螢光顯示屏以便用於座枱的遊戲形式，相等於將一部汽車改變爲一個１／３２比例的模型。通常，模型的汽車窗門未必能夠捲動，水潑沒有眞正的功能和馬達不是內燃式的一種。反過來說，仿效的版本一般缺少了某些特徵。

在座枱的〝食鬼〞遊戲來說，基本上，設計者希望能保存或模倣原著的神韻。不過，現時產品中已犧牲了原版的某些細微之處。而這些地方，正是原來遊戲最具色彩之要點。



6502
80



Coleco Pac-Man.

在兩個座枱式〝食鬼〞遊戲版本中，Coleco的比起Tomy的較具原版特色。其實，Coleco，原是一間專門出產一般迷離電子遊戲機的公司。他們最富盛名的四種電視遊戲複製品計有Pac－Man，Galaxian，Donkey Kong及Frogger等。

遊戲機的螢幕，設有罩蓋掩蔽，藉以阻隔外來的光線。事實上，這個設計，無疑是要使遊戲者具有玩大型遊戲機的感覺。彩色的貼紙，櫃身的卡通彩畫、罩蓋的裝飾和控制鋑板的佈置，都有仿效原初設計的遊戲機。

食鬼遊戲的鋑版控制，包括一個簡單的滑動掣。（SKILL1、OFF、和SKILL2）和兩個四方間的JOYSTICK（歡樂棒）。左邊的棒，是食鬼方向控制器及遊戲選擇掣，當然，遊戲選擇有四種。

首先是遊戲自動進行的示範。但要眞正遊戲時，則可有下列選擇：（1）標準的單人遊戲，（2）兩人比賽食鬼，在這個遊戲中，你本人及你的對手，同時對同一圓點及怪獸作競爭。積分表所顯示，是參賽者得分的差別和誰是領先者。（3）一個新奇變化的遊戲，叫做〝食和走（EAT AND RUN）〞，在這個遊戲中，螢幕出現了四粒頗爲活躍的圓點，和四頭鬼怪，各自保護着一粒圓點。在遊戲過程中，遊戲者要捕捉其中一個活動中的圓點，而當門打開時，將它放進中間的圍欄處。當然，在進行上列行動時，必須把握適當時間——就是於所有怪獸離開保衛位置時，才可進行。無論如何，EAT & RUN是較Pac－Man更爲有趣，況且，Coleco集三種遊戲於一機，必定增添不少姿彩。

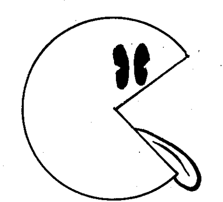
此外螢幕能顯示多個不同形狀的物體，更爲人難以置信。基本上螢光幕本身，是可顯示出三種顏色的；紅、黃及綠。至於第四種顏色，則爲紫色。不過，這顏色是倚靠一塊固定在螢幕面某處上的顏色過濾片來造成的。

要解決在螢光幕上多種不同的角式問題——PAC MAN、鬼怪、能量（ENERGIZER）及跳躍圓點等，螢幕設計者設計出個複右的螢幕角式，當螢光幕完全被照亮時，便現出，PAC－MAN（開口或閉口）及圓點的外形。

不過，螢光幕有一特點，它的複合組成單元，每次只能現出一點或一隻PAC－MAN。當所有的單元被照亮時，只有鬼怪出現。圓點及PAC－MAN黃色的，而鬼怪及四能量點爲紅色。此外，組成迷宮牆的線，亦是紅色。當吃了能量點後，四頭鬼怪便會完失去他們的食鬼能力——相等於原來大型遊戲中的〝爆破〞時間（BLUE TIME）。這時，螢幕便會閃動一段時間，直至鬼怪再次得到動力。

螢幕佈置類似柵格，在柵格之上，兩行爲形勢顯示，包括在遊戲中餘下的PAC－MAN數目、和遊戲完結（〝GAME OVER〞）指示。另外箭咀左右方向則顯示誰爲領先者。最後，在遊戲完畢後，〝BEST〞字會閃動多次，並顯示在遊戲機開動至目前爲止，所獲的最高分數。在遊戲中，五位的分數表，會出現在螢幕的上端。

筆者喜歡Coleco Pac－Man的特性。其中之一，爲迷宮的佈置，它非常接近於原來大型遊戲機的版本。雖然其設計未能盡善盡美，但在遊戲區中佈置的64點／4顆能量點和迷牆的巧妙安排，能使遊戲者控制轉角滑順，樂趣無窮。



6502 80

還有，遊戲機發出的音樂也是它的另一優點。最難得的是這段音樂，保留著原來版本的音韻。此外，遊戲在10，000分時，遊戲者會獲賞一隻PAC－MAN。要達到這個分數，絕非容易。因爲一般的遊戲者，在第一局時，分數往只能得到6－8000分。通常，以Coleco的計分法，一般的遊戲者可以得到2－3000分。

不過，VF顯示的最大弊處，是任何物體，其移動一定是〝跳步〞式的。這種形式的移動，使人難以預測物體的方向性。尤其是在彎角位或十字路口，PAC MAN往往就在這裏死於非命。

另一方面，要提及的是Col－eco遊戲機的噪音，它是屬於一種電子鳴聲。聲音亦非常之大穩定。故此，任何人玩了不久，便會覺得非常煩噪。筆者不明白爲甚麼座枱式遊戲機製造商不設法將問題解決。

這部遊戲機售價，約在＄75美元之下，由於此機不能應用編寫程序，故這價錢未免昂貴一點。但無論如何，筆者估計，這種遊戲機是今年聖誕節禮物中的大熱門。

Tomy Pac－Man

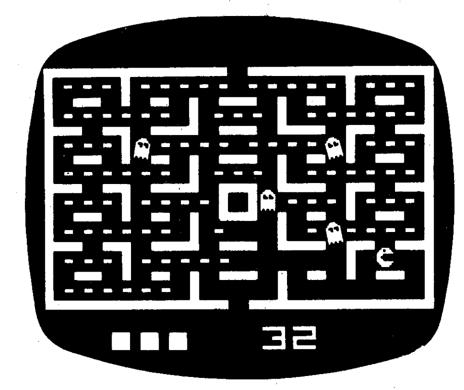
Tomy的產品，外形像一隻飛碟，與原來大型版本，有頗大的分別。

眞空的螢光顯像部份，其概念非常簡單。它能顯示不同的圖像一一如PAC－MAN，鬼怪及圓點等於一個平面擺放的5×8柵格內。在這個設計中，PAC－MAN及鬼怪永遠不能同時佔用同一個位置，而只能在同一柵格內，各自佔有兩邊位置。在一般情形下，遊戲者是不會知道何時才會被附近的鬼怪捕捉的。還有，遊戲者將鬼怪吃掉，亦沒有玩大型遊戲機的感覺。

在這個簡單的迷宮內，總共有18粒圓點和兩粒能量點。控制P－AC－MAN。則利用四個方向性的按掣。其安排很易使人聯想起很多用於小型手提遊戲機中的設計，設有JOYSTICK，只有按掣。不過，最奇妙的部份是當PA－C－MAN從右邊移至左邊時，只能吃掉圓點。筆者懷疑這設計，是簡單迷宮，增加些少趣味。

Tomy遊戲機中的一個較好特點，是設有一個水果（啫哩），在螢幕的方格內出現兩次。如果遊戲的迷宮較爲複雜的話，這個生果必定使遊戲增加不少刺激。因爲吃掉生果的分數實在很多，高達100分。

雖然，Tomy的食鬼遊戲機沒有保留大型機的色彩，但無論如何，其售價僅在＄35美元之下。此外，此機噪音比Coleco遊戲機爲低。對很多小朋友來說，作爲家中的娛樂，此機可作考慮。

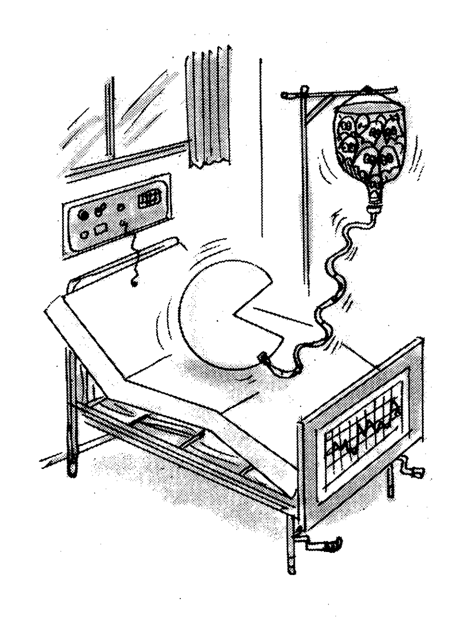


Atari VCS Pac-Man.

電視遊戲版本

當Atari宣佈Pac－Man註冊成爲他們的電視遊戲盒式帶後，擁有VCS和熱愛電視遊戲的人，爲之歡喜若狂。去年，Atari在報章上刊出廣告，鄭重介紹Pac－Man遊戲盒帶後，曾一段時間引起搶購熱潮。

迷宮爲水平擺放，目的是配合家庭電視機的設計。迷宮共有兩個出口，一個處於螢幕上端，一個則處於下端。迷宮的內部結構，可謂迂迴曲折。

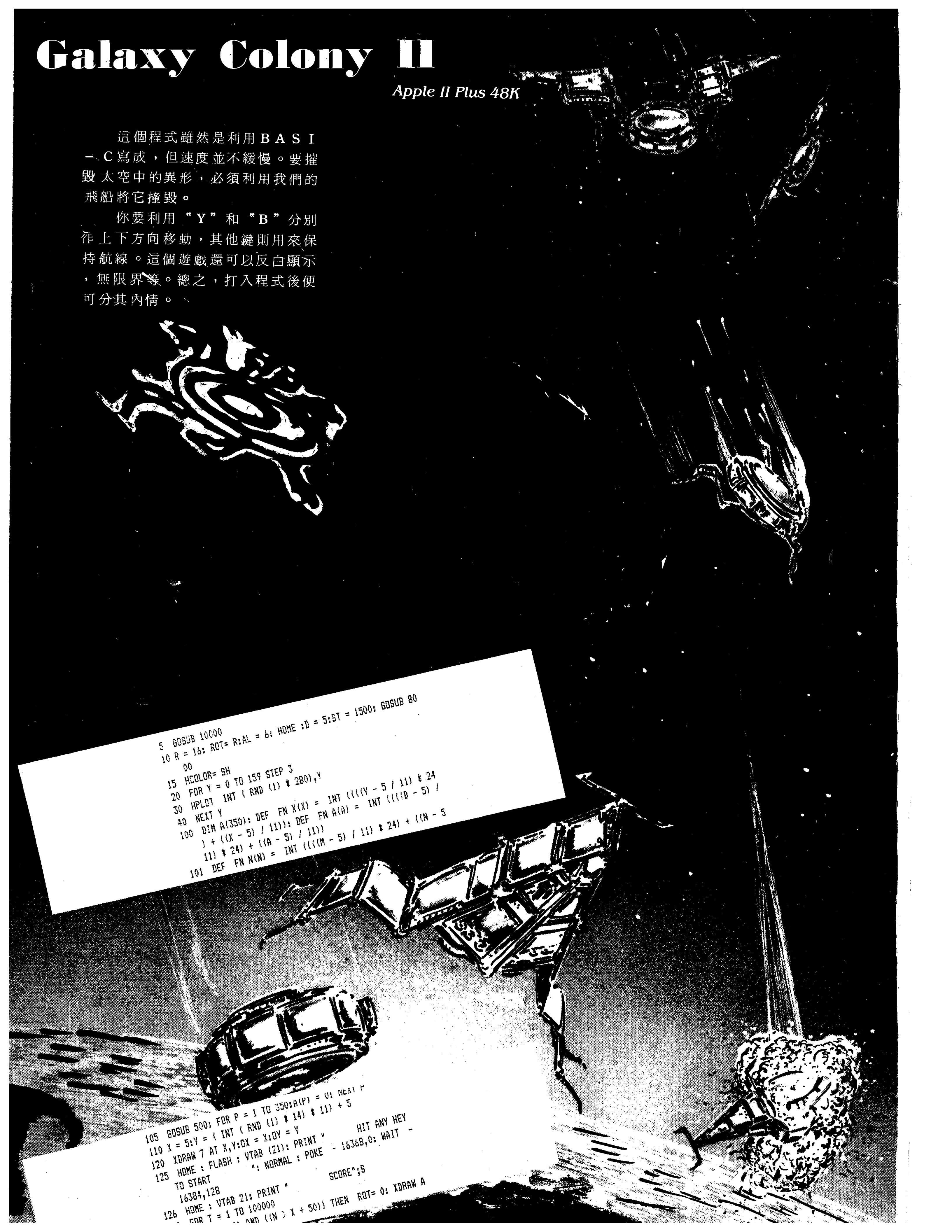


至於和原著不同的地方計有：

所有圓點用破折號代替，水果賞勵改爲黃色及橙色盒子，名稱叫做〝維生素〞。能量圓點亦改爲破折號，叫作〝動力藥丸〞。

此外，Pac－Man在Atari版本中的張口動作，亦與別不同。它是垂直咀嚼，而不是張口大吃。至於怪獸，今回眞正叫做〝鬼〞，它沒有活動的眼睛來指示方向，況且，鬼的顏色全部一樣。但無論如何，當那些食鬼怪吸收了能量後（吃了動力藥丸？），它們會轉爲藍色。每次遊戲開始時，共有四隻Pac－Man生存，假如能夠將螢幕上圓點消去的話，便可得到一額外。

無可否認，市面對Atari Pac－Man的設計者，流傳着很多批評。但無論如何，在電視的迷宮式遊戲中來說，這產品設計頗爲不俗，如沒有原來的大型食鬼遊戲的話，這遊戲盒帶應享有至高的榮譽。



Galaxy Colony II

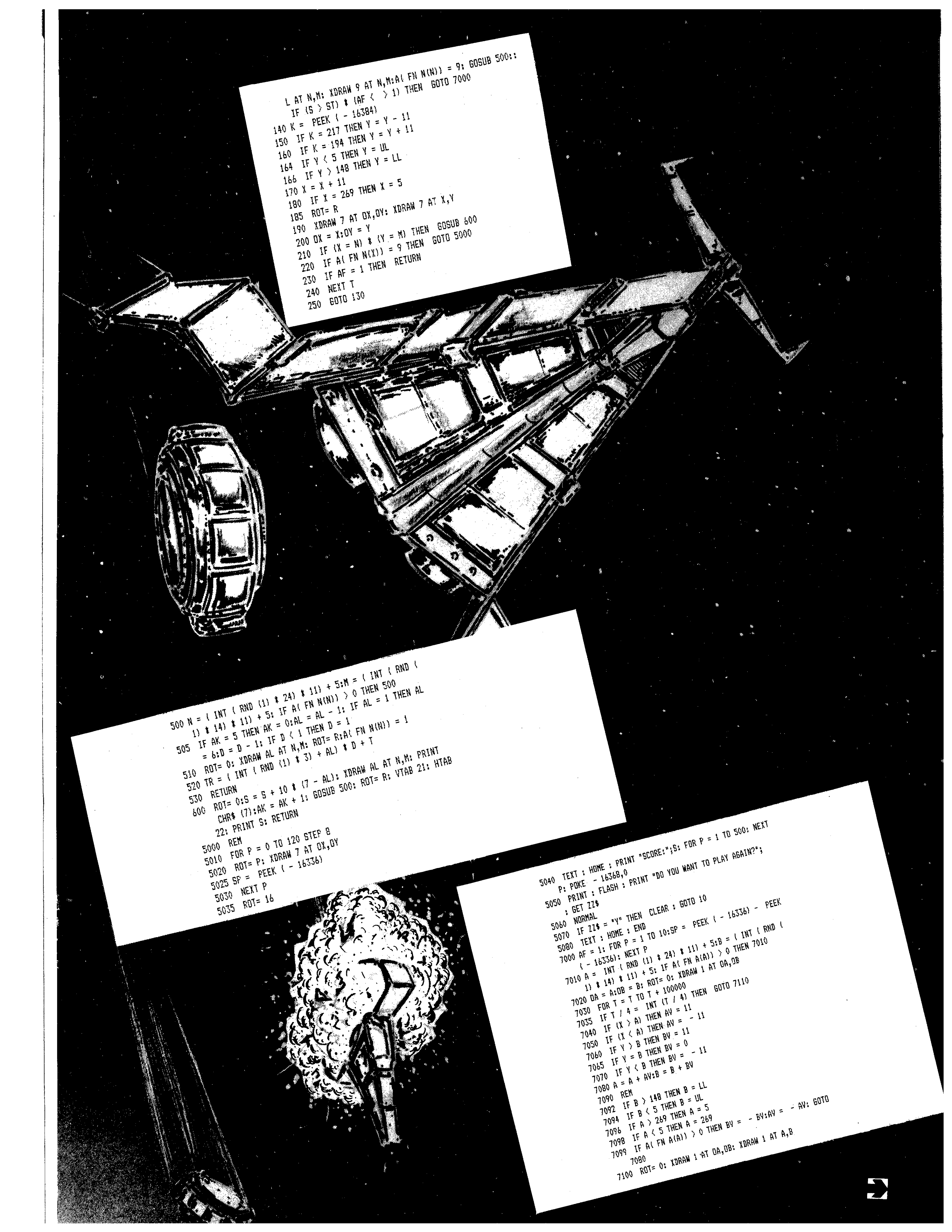
Apple II Plus 48K

這個程式雖然是利用BASI－C寫成，但速度並不緩慢。要摧毀太空中的異形，必須利用我們的飛船將它撞毀。

你要利用“Y”和“B”分別作上下方向移動，其他鍵則用來保持航線。這個遊戲還可以反白顯示，無限界等。總之，打入程式後便可分其內情。

```
5  GOSUB 10000
10 R = 16: ROT= R:AL = 6: HOME :D = 5:ST = 1500: GOSUB 80
    00
15  HCOLOR= SH
20  FOR Y = 0 TO 159 STEP 3
30  HPLOT  INT ( RND (1) * 280),Y
40  NEXT Y
100  DIM A(350): DEF  FN X(X) =  INT ((((Y - 5 / 11) * 24
    ) + ((X - 5) / 11)): DEF  FN A(A) =  INT ((((B - 5) /
    11) * 24) + ((A - 5) / 11))
101  DEF  FN N(N) =  INT ((((M - 5) / 11) * 24) + ((N - 5
```

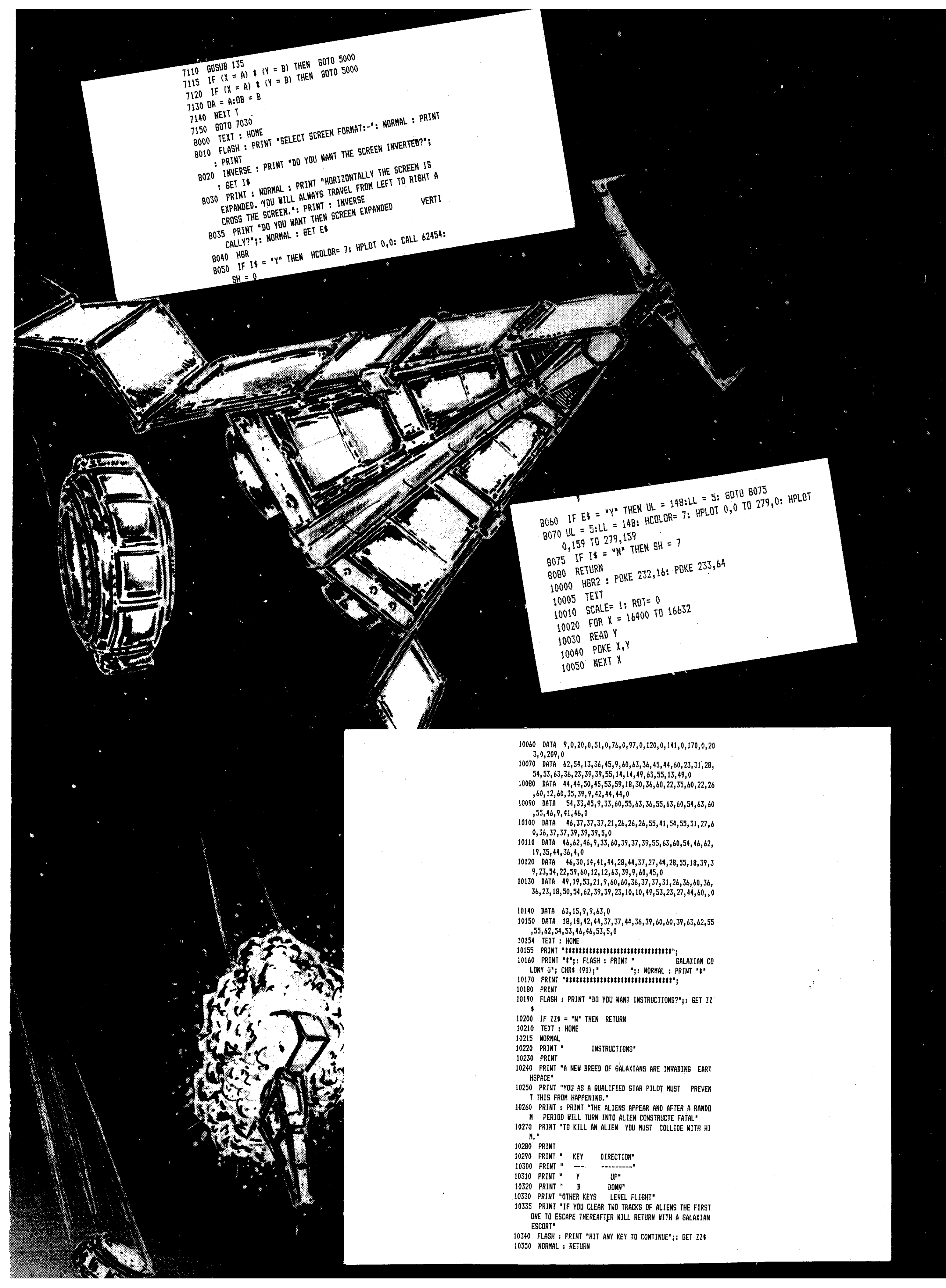
```
                                           = 0: NEXT P
105  GOSUB 500: FOR P = 1 TO 350:A(P) = 0: NEXT P
110 X = 5:Y = ( INT ( RND (1) * 14) * 11) + 5
120  XDRAW 7 AT X,Y:OX = X:OY = Y
125  HOME : FLASH : VTAB (21): PRINT "      HIT ANY KEY
    TO START         ": NORMAL : POKE  - 16368,0: WAIT  -
    16384,128
126  HOME : VTAB 21: PRINT "          SCORE";S
    FOR T = 1 TO 100000
      AND ((N > X + 50)) THEN  ROT= 0: XDRAW A
```



```
    L AT N,M: XDRAW 9 AT N,M:A( FN N(N)) = 9: GOSUB 500::
    IF (S > ST) * (AF <  > 1) THEN  GOTO 7000
140 K =  PEEK ( - 16384)
150  IF K = 217 THEN Y = Y - 11
160  IF K = 194 THEN Y = Y + 11
164  IF Y < 5 THEN Y = UL
166  IF Y > 148 THEN Y = LL
170 X = X + 11
180  IF X = 269 THEN X = 5
185  ROT= R
190  XDRAW 7 AT OX,OY: XDRAW 7 AT X,Y
200 OX = X:OY = Y
210  IF (X = N) * (Y = M) THEN  GOSUB 600
220  IF A( FN N(X)) = 9 THEN  GOTO 5000
230  IF AF = 1 THEN  RETURN
240  NEXT T
250  GOTO 130
```

```
500 N = ( INT ( RND (1) * 24) * 11) + 5:M = ( INT ( RND (
     1) * 14) * 11) + 5: IF A( FN N(N)) > 0 THEN 500
505  IF AK = 5 THEN AK = 0:AL = AL - 1: IF AL = 1 THEN AL
     = 6:D = D - 1: IF D < 1 THEN D = 1
510  ROT= 0: XDRAW AL AT N,M: ROT= R:A( FN N(N)) = 1
520 TR = ( INT ( RND (1) * 3) + AL) * D + T
530  RETURN
600  ROT= 0:S = S + 10 * (7 - AL): XDRAW AL AT N,M: PRINT
     CHR$ (7):AK = AK + 1: GOSUB 500: ROT= R: VTAB 21: HTAB
     22: PRINT S: RETURN
5000  REM
5010  FOR P = 0 TO 120 STEP 8
5020  ROT= P: XDRAW 7 AT OX,OY
5025 SP =  PEEK ( - 16336)
5030  NEXT P
5035  ROT= 16
```

```
5040  TEXT : HOME : PRINT "SCORE:";S: FOR P = 1 TO 500: NEXT
     P: POKE  - 16368,0
5050  PRINT : FLASH : PRINT "DO YOU WANT TO PLAY AGAIN?";
     : GET ZZ$
5060  NORMAL
5070  IF ZZ$ = "Y" THEN  CLEAR : GOTO 10
5080  TEXT : HOME : END
7000 AF = 1: FOR P = 1 TO 10:SP =  PEEK ( - 16336) -  PEEK
     ( - 16336): NEXT P
7010 A =  INT ( RND (1) * 24) * 11) + 5:B = ( INT ( RND (
     1) * 14) * 11) + 5: IF A( FN A(A)) > 0 THEN 7010
7020 OA = A:OB = B: ROT= 0: XDRAW 1 AT OA,OB
7030  FOR T = T TO T + 100000
7035  IF T / 4 =  INT (T / 4) THEN  GOTO 7110
7040  IF (X > A) THEN AV = 11
7050  IF (X < A) THEN AV =  - 11
7060  IF Y > B THEN BV = 11
7065  IF Y = B THEN BV = 0
7070  IF Y < B THEN BV =  - 11
7080 A = A + AV:B = B + BV
7090  REM
7092  IF B > 148 THEN B = LL
7094  IF B < 5 THEN B = UL
7096  IF A > 269 THEN A = 5
7098  IF A < 5 THEN A = 269
7099  IF A( FN A(A)) > 0 THEN BV =  - BV:AV =  - AV: GOTO
     7080
7100  ROT= 0: XDRAW 1 AT OA,OB: XDRAW 1 AT A,B
```
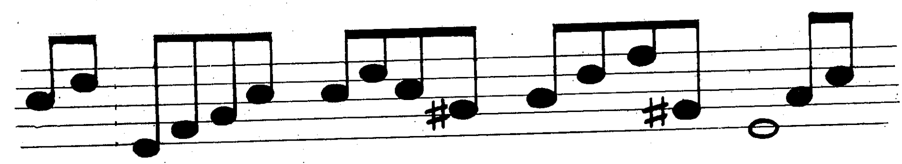


```
7110  GOSUB 135
7115  IF (X = A) * (Y = B) THEN  GOTO 5000
7120  IF (X = A) * (Y = B) THEN  GOTO 5000
7130 OA = A:OB = B
7140  NEXT T
7150  GOTO 7030
8000  TEXT : HOME
8010  FLASH : PRINT "SELECT SCREEN FORMAT:-": NORMAL : PRINT
    : PRINT
8020  INVERSE : PRINT "DO YOU WANT THE SCREEN INVERTED?";
    : GET I$
8030  PRINT : NORMAL : PRINT "HORIZONTALLY THE SCREEN IS
    EXPANDED. YOU WILL ALWAYS TRAVEL FROM LEFT TO RIGHT A
    CROSS THE SCREEN.": PRINT : INVERSE
8035  PRINT "DO YOU WANT THEN SCREEN EXPANDED        VERTI
    CALLY?";: NORMAL : GET E$
8040  HGR
8050  IF I$ = "Y" THEN  HCOLOR= 7: HPLOT 0,0: CALL 62454:
    SH = 0
```

```
8060  IF E$ = "Y" THEN UL = 148:LL = 5: GOTO 8075
8070 UL = 5:LL = 148: HCOLOR= 7: HPLOT 0,0 TO 279,0: HPLOT
    0,159 TO 279,159
8075  IF I$ = "N" THEN SH = 7
8080  RETURN
10000  HGR2 : POKE 232,16: POKE 233,64
10005  TEXT
10010  SCALE= 1: ROT= 0
10020  FOR X = 16400 TO 16632
10030  READ Y
10040  POKE X,Y
10050  NEXT X
```

```
10060  DATA  9,0,20,0,51,0,76,0,97,0,120,0,141,0,170,0,20
    3,0,209,0
10070  DATA  62,54,13,36,45,9,60,63,36,45,44,60,23,31,28,
    54,53,63,36,23,39,39,55,14,14,49,63,55,13,49,0
10080  DATA  44,44,50,45,53,59,18,30,36,60,22,35,60,22,26
    ,60,12,60,35,39,9,42,44,44,0
10090  DATA   54,33,45,9,33,60,55,63,36,55,63,60,54,63,60
    ,55,46,9,41,46,0
10100  DATA   46,37,37,37,21,26,26,26,55,41,54,55,31,27,6
    0,36,37,37,39,39,39,5,0
10110  DATA  46,62,46,9,33,60,39,37,39,55,63,60,54,46,62,
    19,35,44,36,4,0
10120  DATA   46,30,14,41,44,28,44,37,27,44,28,55,18,39,3
    9,23,54,22,59,60,12,12,63,39,9,60,45,0
10130  DATA  49,19,53,21,9,60,60,36,37,37,31,26,36,60,36,
    36,23,18,50,54,62,39,39,23,10,10,49,53,23,27,44,60,,0

10140  DATA  63,15,9,9,63,0
10150  DATA  18,18,42,44,37,37,44,36,39,60,60,39,63,62,55
    ,55,62,54,53,46,46,53,5,0
10154  TEXT : HOME
10155  PRINT "********************************";
10160  PRINT "*";: FLASH : PRINT "           GALAXIAN CO
    LONY ü"; CHR$ (91);"         ";: NORMAL : PRINT "*"
10170  PRINT "********************************";
10180  PRINT
10190  FLASH : PRINT "DO YOU WANT INSTRUCTIONS?";: GET ZZ
    $
10200  IF ZZ$ = "N" THEN  RETURN
10210  TEXT : HOME
10215  NORMAL
10220  PRINT "        INSTRUCTIONS"
10230  PRINT
10240  PRINT "A NEW BREED OF GALAXIANS ARE INVADING  EART
    HSPACE"
10250  PRINT "YOU AS A QUALIFIED STAR PILOT MUST   PREVEN
    T THIS FROM HAPPENING."
10260  PRINT : PRINT "THE ALIENS APPEAR AND AFTER A RANDO
    M   PERIOD WILL TURN INTO ALIEN CONSTRUCTE FATAL"
10270  PRINT "TO KILL AN ALIEN  YOU MUST  COLLIDE WITH HI
    M."
10280  PRINT
10290  PRINT "   KEY     DIRECTION"
10300  PRINT "   ---     ---------"
10310  PRINT "    Y         UP"
10320  PRINT "    B        DOWN"
10330  PRINT "OTHER KEYS    LEVEL FLIGHT"
10335  PRINT "IF YOU CLEAR TWO TRACKS OF ALIENS THE FIRST
    ONE TO ESCAPE THEREAFTER WILL RETURN WITH A GALAXIAN
    ESCORT"
10340  FLASH : PRINT "HIT ANY KEY TO CONTINUE";: GET ZZ$
10350  NORMAL : RETURN
```

美妙的音符



　　你會經嘗試過在Apple裏編寫一些音樂的程式嗎？你希望找出發出音樂的技巧嗎？

　　現在你不必再費心思了。"美妙的音符"替你開拓出Apple發聲的秘密。

　　這個程式裏包括六首歌仔，其中一首神秘歌。筆者肯定這首歌你會聽過，不過歌名就無從稽考。選擇5，你便會聽到一些很短的聲音，按下P試試。你便會聽到一些較長的聲音。再按下試試。你會發覺每次按下一次"P"的時候，NOTE LENGTH（音調長度）便會增加一，看看按下多少次P才能眞正聽到這首歌曲？

　　除此之外，還有個音樂打字機（MUSICAL TYPEWRITER）都是十分有趣。如果你對於音樂產生仍然不大清楚，按下"I"便會明白了！

　　這個程式的各個主要部都加上了REM語句，相信大家是會很容易明瞭的。



```
10   REM  ***********************
20   REM  **                   **
25   REM  ** WONDEFUL MELODY   **
26   REM  **                   **
27   REM  **  EDITED    BY     **
28   REM  **                   **
29   REM  **     OLD MAN       **
30   REM  **                   **
32   REM  ***********************
100  GOSUB 60000: REM MUSIC POKES
150  GOTO 700: REM GOTO MENU
```

PLAY A NOTE
===========

```
400  POKE 0,255 - PI: REM PITCH
401  POKE 1,LE: REM NOTE LENGTH
402  CALL 771: REM PLAY THE NOTE
403  RETURN
499  REM
200  IF REST THEN  HTAB 12: PRINT "<REST=";REST;">": FOR
    X = 1 TO REST: NEXT :REST = 0: RETURN
300  PRINT "PITCH:";PI;",";: HTAB 12: PRINT "LENGTH:";LE
399  REM
```

MENU
====

```
500  PRINT :A$ = "----------": PRINT A$: PRINT "ANY KEY:": PRINT
    A$;: VTAB  PEEK (37): GET A$
700  TEXT : NORMAL :DOT$ = "................................
    ...........": FOR I = 1 TO 23: VTAB I: HTAB 1: PRINT
    DOT$;: NEXT : PRINT  LEFT$ (DOT$,39);: POKE 2039, ASC
    (".") + 128
800  POKE 32,5: POKE 33,35: VTAB 3: HTAB 5: INVERSE : PRINT
    " WONDERFUL MELODY ": NORMAL : PRINT : PRINT "<1> CHA
    RGE": PRINT "<2> SHAVE AND A HAIRCUT": PRINT "<3> HER
    E COMES THE BRIDE": PRINT "<4> BUGLE CALL"
900  PRINT "<5> ": PRINT "<6> C?-SCALE": PRINT "<7> MUSIC
    AL TYPEWRITER": PRINT "<8> RANDOM NOTES": PRINT "<9>
    SLIDE WHISTLE": PRINT : PRINT "<I> INSTRUCTIONS": PRINT
    : PRINT "<Q> QUIT"
1000  VTAB 22: HTAB 5: PRINT "SELECT:";: FLASH : PRINT "
    ": NORMAL :HT = 1:A$ = " MYSTERY SONG."
1100  FOR I = 1 TO 3:PI =  INT ( RND (1) * 200) + 25:LE =
    20: GOSUB 400: NEXT
1200 HT = HT + 1 - 12 * (HT = 12): VTAB 9: HTAB HT + 4: PRINT
    "S";: HTAB HT + 3: PRINT  MID$ (A$,HT,1):K =  PEEK ( -
    16384): IF K < 128 THEN 1200
1300  POKE  - 16368,0: IF (K < 177 OR K > 185) AND K <  >
    209 AND K <  > 201 THEN 1000
1400  VTAB 9: HTAB 5: PRINT "MYSTERY SONG"
1500  VTAB 22: HTAB 12: PRINT  CHR$ (K): TEXT : IF K = 20
    9 THEN 13600
1505  IF K = 201 THEN 50010
1600  ON K - 176 GOTO 1800,2700,3800,9900,5700,11800,9500
    ,5100,5300: GOTO 700
1700  REM
```

CHARGE SONG
===========

```
1800  HOME :PI = 63:LE = 40: GOSUB 200
1900 PI = 111: GOSUB 200
2000 PI = 141: GOSUB 200
2100 PI = 159:LE = 50: GOSUB 200
2200 REST = 100: GOSUB 200
2300 PI = 141:LE = 60: GOSUB 200
2400 PI = 159:LE = 255: GOSUB 200
2500  GOTO 500
2600  REM
```

```
SHAVE & HAIRCUT
===============
2700  HOME :PI = 103:LE = 100: GOSUB 200: REM SHAVE
2800 PI = 63:LE = 50: GOSUB 200 REM AND
2900  GOSUB 200: REM A
3000 PI = 84:LE = 90: GOSUB 200: REM HAIR-
3100 PI = 63:LE = 60: GOSUB 200: REM CUT,
3200 REST = 300: GOSUB 200
3300 PI = 103:LE = 80: GOSUB 200: REM SIX
3400 REST = 22: GOSUB 200
3500 PI = 111: GOSUB 200: REM BITS!
3600  GOTO 500
3700  REM
HERE COMES THE BRIDE
====================
3800  HOME :PI = 63:LE = 255: GOSUB 200: REM HERE
3900 PI = 111:LE = 180: GOSUB 200: REM COMES
4000 REST = 20: GOSUB 200
4100 LE = 80: GOSUB 200: REM THE
4200 LE = 255: GOSUB 200: REM BRIDE;
4300 REST = 350: GOSUB 200
4400 PI = 63:LE = 255: GOSUB 200: REM ALL
4500 PI = 127:LE = 180: GOSUB 200: REM FAT
4600 REST = 3: GOSUB 200
4700 PI = 103:LE = 80: GOSUB 200: REM AND
4800 PI = 111:LE = 255: GOSUB 200: REM WIDE!
4900  GOTO 500
5000  REM
RANDOM
======
5100  FLASH : FOR I = 1 TO 20:PI =  INT ( RND (1) * 255) +
     1:LE =  INT ( RND (1) * 30) + 10: POKE 0,PI: POKE 1,L
     E: CALL 771: VTAB  INT ( RND (1) * 24) + 1: HTAB  INT
     ( RND (1) * 39) + 1: PRINT  CHR$ ( INT ( RND (1) * 64
     ) + 32);: NEXT :LE = 255: POKE 0,PI: POKE 1,LE: CALL
     771: NORMAL : GOTO 700
5200  REM

SLIDE WHISTLE
=============
5300 COL$ = "::::::::::::::::::::::::::::::::::::::::": FOR
     I = 1 TO 23: VTAB I: HTAB 1: PRINT COL$;: NEXT : PRINT
      LEFT$ (COL$,39);: POKE 2039, ASC (":") + 128
5400 LE = 4: FOR PI = 254 TO 1 STEP  - 2: POKE 0,255 - PI
     : POKE 1,LE: CALL 771: NEXT
5500  FOR PI = 1 TO 50: POKE 0,255 - PI * 5: POKE 1,LE: CALL
     771: NEXT : GOTO 700
5600  REM

MYSTERY SONG
============
5700  HOME : PRINT "MYSTERY SONG!": PRINT "------------":
      PRINT : PRINT "THIS SONG HAS BEEN COMPRESSED. EACH T
     IMEIT IS PLAYED, THE NOTES GET LONGER.     SEE HOW FA
     ST YOU CAN NAME THE SONG."
```

```
5800  FOR LX = 2 TO 80 STEP 2
5900  VTAB 10: HTAB 1: CALL  - 958: PRINT "NOTE LENGTH: "
     ;LX / 2;" (NORMAL IS 40)"
6000 PI = 111:LE = LX: GOSUB 400: REM YAN-
6100  GOSUB 400: REM KEE
6200 PI = 127: GOSUB 400: REM DOO-
6300 PI = 141: GOSUB 400: REM DLE
6400 PI = 111: GOSUB 400: REM CAME
6500 PI = 141: GOSUB 400: REM TO
6600 PI = 127: GOSUB 400: REM TOWN
6700 PI = 63: GOSUB 400: REM A-
6800 PI = 111: GOSUB 400: REM RID-
6900  GOSUB 400: REM ING
7000 PI = 127: GOSUB 400: REM ON
7100 PI = 141: GOSUB 400: REM A
7200 PI = 111:LE = LX * 2: GOSUB 400: REM PO-
7300 PI = 103:LE = LX: GOSUB 400: REM NY.
7400 PI = 63: GOSUB 400: REM HE
7500 PI = 111: GOSUB 400: REM STUCK
7600  GOSUB 400: REM A
7700 PI = 127: GOSUB 400: REM FEA-
7800 PI = 141: GOSUB 400: REM THER
7900 PI = 147: GOSUB 400: REM IN
8000 PI = 141: GOSUB 400: REM HIS
8100 PI = 127: GOSUB 400: REM HAT
8200 PI = 111: GOSUB 400: REM AND
8300 PI = 103: GOSUB 400: REM CALLED
8400 PI = 63: GOSUB 400: REM IT
8500 PI = 84: GOSUB 400: REM RICE-
8600 PI = 103: GOSUB 400: REM A-
8700 PI = 111:LE = 2 * LX: GOSUB 400: REM RO-
8800 LE = LX: GOSUB 400: REM NI!
8900  IF LX = 80 THEN 700
9000  VTAB 12: PRINT "<P> PLAY AGAIN WITH LONGER NOTES.":
     PRINT "<G> GIVE UP; PLAY AT NORMAL SPEED.": PRINT "<
    Q> QUIT PLAYING THAT SONG... NOW!": PRINT : PRINT "
     SELECT:";: GET A$: PRINT A$: IF A$ <  > "Q" AND A$ <
     > "P" AND A$ <  > "G" THEN 9000
9100  IF A$ = "P" THEN  NEXT
9200  IF A$ = "G" THEN LX = 80: GOTO 5900
9300  GOTO 700
9400  REM
MUSICAL TYPEWRITER
========================
9500  HOME : PRINT "     EACH KEY TYPES A DIFFERENT NOTE."
    : HTAB 13: PRINT "<CTRL-Q> TO QUIT.": PRINT "----------
    ----------------------------------------"
9600  GET A$: PRINT A$;:PI =  ASC (A$): POKE 0,PI * 2: POKE
    1,20: CALL 771: IF A$ <  >  CHR$ (17) THEN 9600
9700  GOTO 700
9800  REM
BUGLE CALL
==========
9900  HOME : VTAB 24:BT = 72: FOR I = 1 TO 2
10000 REST = 50: GOSUB 200
```

```
10100 PI = 63:LE = BT / 3: GOSUB 200
10200 FOR J = 1 TO 3
10300 PI = 111:LE = BT / 3: GOSUB 200
10400 REST = 5: GOSUB 200
10500 PI = 141:LE = BT / 4: GOSUB 200
10600 PI = 111:LE = BT / 4: GOSUB 200
10700 PI = 63:LE = BT / 4: GOSUB 200
10800 REST = 1: GOSUB 200
10900  IF I = 2 AND J = 3 THEN  GOSUB 200:REST = 15: GOSUB
    200:PI = 111:LE = 2 * BT: GOSUB 200: GOTO 11500
11000  IF J = 3 THEN PI = 111:LE = BT / 2: GOSUB 200
11100 PI = 141:LE = BT / 3 + BT * (J = 3): GOSUB 200
11200  IF J < 3 THEN REST = 15: GOSUB 200
11300  NEXT J
11400 PI = 111:LE = BT / 2: GOSUB 200
11500  NEXT I
11600  GOTO 500
11700  REM

C-SCALE
=======
11800  HOME :A$ = "CDEFGABCBAGFEDC": FOR I = 1 TO  LEN (A
    $): VTAB I: HTAB 23: PRINT "("; MID$ (A$,I,1);")": NEXT
    : VTAB 1
11900 PI = 63:LE = 80: GOSUB 200
12000 PI = 84: GOSUB 200
12100 PI = 103: GOSUB 200
12200 PI = 111: GOSUB 200
12300 PI = 127: GOSUB 200
12400 PI = 141: GOSUB 200
12500 PI = 153: GOSUB 200
12600 PI = 159: GOSUB 200
12700 PI = 153: GOSUB 200
12800 PI = 141: GOSUB 200
12900 PI = 127: GOSUB 200
13000 PI = 111: GOSUB 200
13100 PI = 103: GOSUB 200
13200 PI = 84: GOSUB 200
13300 PI = 63:LE = 150: GOSUB 200
13400  GOTO 500
13500  REM

QUIT
====
13600  FOR V = 1 TO 23:H = 25 - V:A = 128 * V + H - (984 *
    INT ((V - 1) / 8)) + 895:B = A + 1:PA =  PEEK (A):PB
    =  PEEK (B): POKE A,221: POKE B,96: POKE 0,10 * V: POKE
    1,9: CALL 771: FOR J = 1 TO 25: NEXT : POKE A,PA: POKE
    B,PB: NEXT : VTAB 23: END
50000  REM

INSTRUCTIONS
============
50010  HOME : PRINT "  C    D    E    F    G    A    B
    C": INVERSE : FOR I = 1 TO 40: PRINT "$$$$$     ";: NEXT

50020  NORMAL : FOR I = 1 TO 6: FOR J = 5 TO 30 STEP 5: VTAB
    I + 1: HTAB J: IF J < > 15 THEN  PRINT "  ";
50030  NEXT : HTAB 40: PRINT " ";: NEXT
50040  VTAB 4:A$ = "  X111X1": GOSUB 50080:A$ = "79X134X6
    ": GOSUB 50080:A$ = "34X947X9": GOSUB 50080
50050 I = 0:A$ = " 63": GOSUB 50060:A$ = " 84": GOSUB 500
    60:A$ = "103": GOSUB 50060:A$ = "111": GOSUB 50060:A$
    = "127": GOSUB 50060:A$ = "141": GOSUB 50060:A$ = "1
    53": GOSUB 50060:A$ = "159": GOSUB 50060: GOTO 50100
50060  INVERSE :I = I + 1: VTAB 10: HTAB I * 5 - 3: PRINT
    A$: POKE 0,255 -  VAL (A$): POKE 1,80: CALL 771: RETURN

50070  FOR I = 1 TO 8: HTAB I * 5 - 2: POKE 50,63 + 192 *
    ( PEEK (37) > 55): PRINT  MID$ (A$,I,1);: NEXT : PRINT
    : RETURN
50080  FOR I = 1 TO 8: HTAB I * 5: IF  MID$ (A$,I,1) < >
    "X" THEN  PRINT  MID$ (A$,I,1);
50090  NEXT : RETURN
50100  NORMAL : VTAB 13: PRINT "TO MAKE MUSIC, ADD LINE 6
    0000 TO YOUR  PROGRAM AND MAKE YOUR FIRST LINE BE---"
    : PRINT : PRINT "  10  GOSUB 60000": PRINT
50110  PRINT "FOR EACH NOTE, DO TWO POKES AND A CALL---
                 -----------          ----"

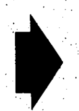
50120  PRINT "  POKE 0,255-PI (PI=PITCH, SEE ABOVE)      P
    OKE 1,LE       (LE=LENGTH, 0-255)          CALL 771
    (PLAYS THE NOTE)";: VTAB 24: HTAB 1: PRINT "ANY KEY:"
    ;: GET A$: GOTO 700
59999  REM

MUSIC POKES
===========
60000  POKE 771,173: POKE 772,48: POKE 773,192: POKE 774,
    136: POKE 775,208: POKE 776,4: POKE 777,198: POKE 778
    ,1: POKE 779,240: POKE 780,8: POKE 781,202: POKE 782,
    208: POKE 783,246: POKE 784,166: POKE 785,0: POKE 786
    ,76: POKE 787,3: POKE 788,3: POKE 789,96: RETURN
```

錄音帶程式保護導論（一）

雖然軟性磁碟幾乎成爲了微電腦的標準外圍貯存媒介（secondary or auxillary storage），但利用錄音帶作外圍貯存媒介，在本地也十分流行。

錄音帶的好處在於它的價錢便宜，而且具有相當程度的可靠性。例如：APII的翻版遊戲軟件，通通是利用卡式錄音帶作爲貯存媒介，因此售價方面十分廉宜。雖然這些翻版錄音軟件是這樣的便宜，但却是被保護的。換句話說，我們不可直接把程式讀入後，把程式的內容，收錄在另一盒錄音帶內。商人應用的保護方法是幾種的，筆者今期所介紹的是其中最簡單的一種。





寫入ROUTINE

在Applesoft Rom內是有兩個主要Routine分別用來處理卡式錄音帶的讀出（READ）和寫入（WRITE）操作的。寫入Routine的進入點在$FECD（十六進數）或65529（−307）（十進位數）。在未進入此ROUTINE前，我們首先要指明寫入操作的開始地址（2個byte），把它們放在$3C和$3D內，而把最後的地址放在$3E和$3F內。

例如：我們想把某段的記憶內容貯存在卡式帶內（300至3D0）。最簡單（平時常用）的方法就是在監視程式（Monitor）內輸入以下的命令：

300．3D0W<CR>

<CR>代表CARRIAGE RUTURN，即按下RETURN的意思。

跟着，按下卡式機的REC和PLAY，當電腦發出BEEP一聲後，便表示錄音機已將那段記憶內容收錄完畢。

另一個途徑，就是根據上文先在$3C、$3D$3E和$3F寫入適當的內容，然後跳往$FEFD。要達到這個目的，我們只要寫入一段很短的機械或滙編語言（Assembly Language）程式便行了。譬如，我們在$300開始我們的程式：

```
0800-   A9 00       LDA   #$00
0802-   85 3C       STA   $3C
0804-   A9 03       LDA   #$03
0806-   85 3D       STA   $3D
0808-   A9 D0       LDA   #$D0
080A-   85 3E       STA   $3E
080C-   A9 03       LDA   #$03
080E-   85 3F       STA   $3F
0810-   4C CD FE    JMP   $FECD
0813-   FF          ???
0814-   FF          ???
```

當輸入完畢後，我們便可以將它執行。很簡單，只要打入在打入300G，然後按下REC和PLAY，便可將$3D0至$3FB的內容貯存在卡式錄音帶裏。

讀出Routine

讀出ROUTINE和寫入ROUTINE的工作原理沒有多大的分別，都是大同小異的。它的進入點在$FEFD，或65277（−259）。和寫入ROUTINE的情況一般，在未進入這個ROUTINE前，首先要把開始地址貯存在$3E和$3F裏。監視程式便會從錄音帶讀入數據，根據指定的記憶位置貯存在記憶內。除了負責讀入數據外，監視程式還進行數據的核對和（CHECK−SUU）工作——讀入的數據會與零頁（Zero Page）的$2E進行一個Exclusive Or的邏輯運算。當最後的記憶位置內容讀入後，監視程式便會讀取多一個byte，看看它的內容是否和$2E相同。如果是相同的話，監視程式便會發出BEEP的聲響，然後返回監視程式。如果不相同的話，電腦便會在發出BEEP聲響前，在螢幕上打出ERR表示錯誤讀入。

看完了上述的解釋，大家可能對讀出和寫入操作已有了一個明確的概念，或許錄音帶的保護方法也找出了一些媚目。



好了，言歸正傳讓我們看看30．FFR是一個怎樣的保護方法。這種保護方法在早期的翻版錄音帶軟件裏。大家可能也曾購買過。但是，大家可有想過載入長度和時間的比例關係呢！那些錄音帶通常是這樣寫着的：「CALL−151 *30．FFR」意思是說，先打入CALL−151跳進監視程式內，然後輸入30．FFR命令，然後按下PLAY。

要將整個程式完全載入通常是需要三至四分鐘時間的。但是回頭一想。$30至$FF只有208個byte（讀入208個byte的時間啊！

解釋保護

難道30．FFR是一個假命令來的？非也。難道30．FFR命令執行完畢後，還有另一段程式讀入嗎？對了！！對了！！錄音帶是分爲兩節的。

筆者曾經說過，讀出內容前，我們必須在$3C、$3D、$3E和$3F內放置適當數值才能夠跳往ROM內的讀出ROUTINE工作。然而，這四個地址剛好$30至$FF的範圍裏。換句話說，要知道眞正程式（第二段）的開始和結尾位置，就非要知道$3C、$3D、$3E和$3F的內容不可。

```
1050-   A9 F5       LDA   #$F5
1052-   85 3C       STA   $3C
1054-   A9 03       LDA   #$03
1056-   85 3D       STA   $3D
1058-   A9 FF       LDA   #$FF
105A-   85 3E       STA   $3E
105C-   A9 91       LDA   #$91
105E-   85 3F       STA   $3F
1060-   A9 00       LDA   #$00
1062-   85 36       STA   $36
1064-   A9 40       LDA   #$40
1066-   85 37       STA   $37
1068-   A9 00       LDA   #$00
106A-   85 00       STA   $00
106C-   A9 04       LDA   #$04
106E-   85 01       STA   $01
1070-   A0 00       LDY   #$00
1072-   A9 A0       LDA   #$A0
1074-   91 00       STA   ($00),Y
1076-   E6 00       INC   $00
1078-   D0 FA       BNE   $1074
107A-   E6 01       INC   $01
107C-   A6 01       LDX   $01
107E-   E0 08       CPX   #$08
1080-   90 F2       BCC   $1074
1082-   AD 56 C0    LDA   $C056
1085-   AD 52 C0    LDA   $C052
1088-   EA          NOP
1089-   EA          NOP
108A-   EA          NOP
108B-   4C FD FE    JMP   $FEFD
108E-   EA          NOP
108F-   EA          NOP
1090-   EA          NOP
1091-   EA          NOP
1092-   EA          NOP
1093-   EA          NOP
1094-   EA          NOP
1095-   EA          NOP
1096-   EA          NOP
```

如果將程式從＄３０開始讀起，是永遠不能夠知道它們的內容的。因爲從這裏載入，第一段程式（或稱ＨＥＡＤＥＲ）是會自動運算（ａｕｔｏｒｕｎ）的。所以唯一的方法就是把錄音帶程式載往另一個記憶位置。譬如，我們可以把程式移至１０３０開始。那麼，我們所發下的命令便是１０３０．１０ＦＦＲ，而不再是３０．ＦＦＲ。這一次，你很快便會聽到ＢＥＥＰ一聲。當聽到ＢＥＥＰ的聲響後，立即停止卡式錄音機繼續運行，按下ＰＡＵＳＥ或ＳＴＯＰ。不要按動任何掣，讓錄音機休息一會兒好嗎？

我們發覺在＄１０３０至＄１０ＦＦ中，只有某些部份可以被內置的ｄｉｓａｓｓｅｍｂｌｅｒ反滙編（ｄｉｓａｓｓｅｍｂｌｅ）的。（在這裏是＄１０５０）

很明顯，被反滙編的那段記憶才是眞正的程式。仔細看看這段程式是執行些什麽工作！噢！明白了！我們可以從這裏知道＄３Ｃ＄，＄３Ｄ，＄３Ｅ和＄３Ｆ的內容。它們正是第二段（眞正程式內容）所需要的資料。可是＄３６和＄３７則是程式的運算的進入點。通常＄３６和＄３７最載着的內容是＄３Ｃ和＄３Ｄ所載着的是一模一樣的。

從程式裏，我們可以知道
開始地址：＄３Ｆ５
結尾地址：＄９１ＦＦ
程式進入點：＄４０００

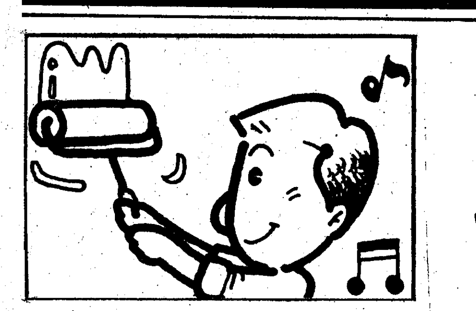
現在讓我們把第二段讀出來：

３Ｆ５．９１ＦＦ＜ＣＲ＞

然後按下錄音機的ＰＬＡＹ。當電腦發出ＢＥＥＰ的聲響後，便表示已電腦已把程式載入。現在你可以執行程式，或者將這個軟件做一份ＢＡＣＫ—ＵＰ。如果你在載入錄音帶前，曾經ＢＯＯＴ了ＤＯ—Ｓ３．３，你便可以把程式貯存在磁碟裏。只要打入ＢＳＡＶＥ　ＦＩＬＥ　ＮＡＭＥ，Ａ＄３Ｆ５，Ｌ＄７ＦＦＦ。在這裏，大家要注意一點，ＤＯＳ３．３在一般情況下，只能夠接受長度＄７ＦＦＦ的檔案，超過這個範圍便會出現Ｅ—ＲＲＯＲ。原因很簡單，因爲Ｆ—ＩＬＥ　ＢＵＦＦＥＲ不夠大。在這裏，＄３Ｆ５至＄９１ＦＦ的長度是相等於＄８４０Ａ，因此上面的命令是不成立的。唯一的方法就是把程式分成兩個檔案貯存。

首先，ＢＳＡＶＥ　ＦＩＬＥ—ＮＡＭＥ，Ａ＄３Ｆ５，Ｌ＄７ＦＦＦ。然後ＢＳＡＶＥ　ＦＩＬ—ＥＮＡＭＥ２，Ａ＄８０００，Ｌ＄Ｅ０Ａ，便大功告成。

如果你沒有磁碟機，又想做Ｂ—ＡＣＫ—ＵＰ，就只有把程式貯存在另一盒錄音帶裏。打入３Ｆ５．９１ＦＦ　Ｗ而當運算時，只要打入４０００Ｇ，程式便會執行呢！◆





滾墨(roller)遊戲

前　言

這個程序是我與正在學電腦的G－SUN先生合作的，許多地方可能還不成熟，若有不當之處，敬請原諒。

遊戲的內容

什麼叫做滾墨（roller）呢？即在圖面上的90％以上地方塗上墨的遊戲。已塗上墨的地方規定不能再塗第二次。用此滾墨可以用來追逐妖怪。

一按下RUN鍵，就會在屏幕上出現遊戲的玩法說明。看誰能用最少的次數做完此遊戲。

請你再按下KEY鍵，此遊戲就開始了，但對於如何安排讀出需要一定熟悉的時間。

稍等片刻之後，由於圖象的偏右邊比中間少了一些滾墨，請你操作KEY鍵。這時，你就可以清除圖象，來消滅一隻妖怪。通過5次的清除之後，妖怪的活動就會加激。即運動的速度更加快。若你三次被妖怪抓住，那麼就算爲GAME OVER，要再等10秒鐘，再重新按下KEY，就可以再玩這遊戲了。

假如你有辦法，在一次不重複的過程中把妖怪清除了，那麼你就能得到10倍的賞分，並用劃線來計算。

順便說明一下，這裏的最高獎賞分數爲23500分。

提高等級

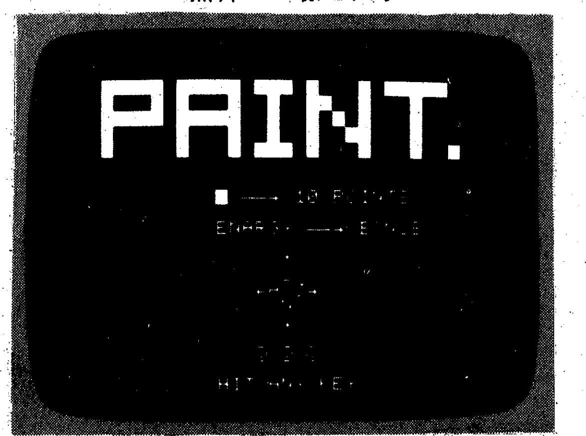
若能很好地應用BEEP指令，就能得到更加滿意的效果。

由於我們這裏是採用BASIC語言編寫的，故此遊戲很容易移植到別種類型的電腦中。

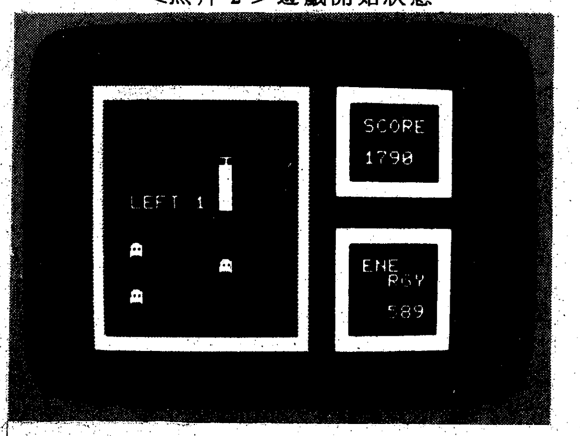
結語

讀取配列需要一定的時間，若下功夫去鑽研，就可更快地掌握。

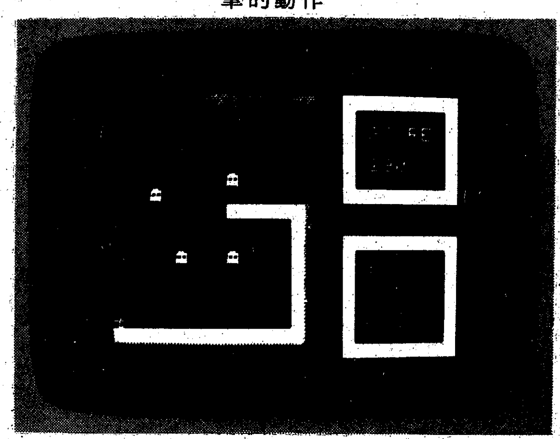
有關這些，請大家下點功夫吧！



＜照片1＞標題圖象



＜照片2＞遊戲開始狀態



＜照片3＞連續不斷地操作讓滾墨筆的動作

CHECK FLAG

Dr．D：

遊戲的內容比較簡單，但玩起來却非常有意思。由於是應用了JR－100的程序，所用的字符還是比較少的。

編輯先生：更進一步要求，能發出聲音就更理想了。

Dr．D：確實，若配上聲音就更理想，在我們製作的原稿中也寫進去了，但花了很大的氣力，其所得到的聲音還很不理想。

編輯先生：不過，這還有個不太理想的地方。我想，若妖怪遠離原來的地方，幸而未被塗掉，那些妖怪一定會很開心的。

Dr．D：啊！是呀！那還是不夠理想的地方。我們想利用有趣的文字加上聲音解說。那麽，這樣大家玩起來就更加有趣了。

編輯先生：有關以上諸點，可以更進一步地改善，那一定可作出更加有趣的程序來了。

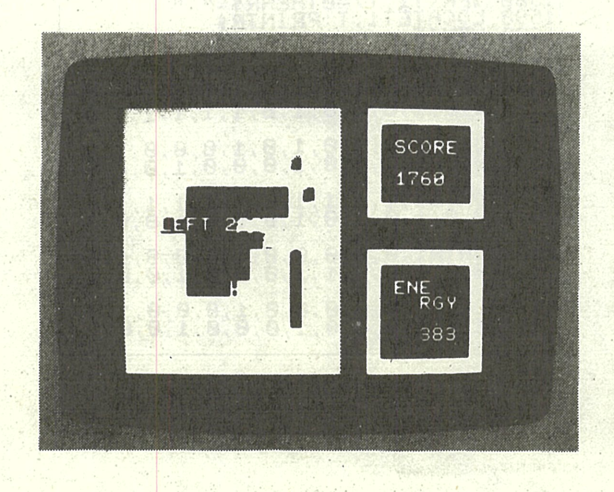
表　二

變量清單

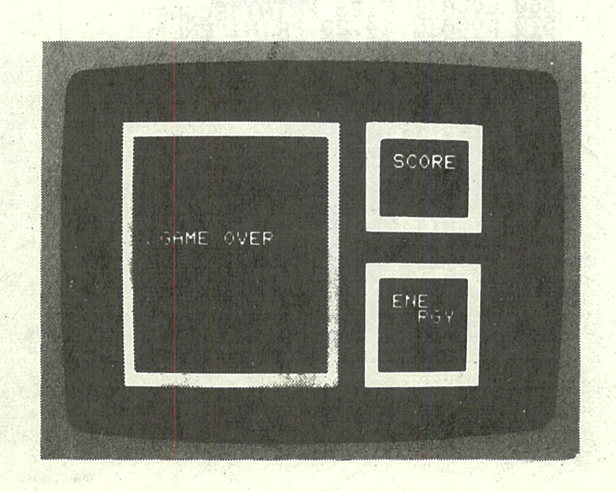
- S3——1幅圖案中的劃線
- E——能級
- P1——圖案的數目
- S——劃線
- Z——滚墨的數目
- V，W1——妖怪的移動量
- A，B——滚墨的位置座標
- Y（t），X（t）——妖怪的位置注標
- P（，）——行動區域
- M（，）——妖怪的位置注標
- L（，）——塗墨的點子
- N（，）——不消去塗墨地方的配列

表　一

	程序的內容
10	配列說明
120～290	遊戲的說明
300～490	初始狀態設定，讀取配列
500～700	作成圖面
710～850	表示敵方（對方）的位置
860～970	滚墨的位置表示
980～1030	劃線點，能級的計數
1070～1120	表示被抓到的情景
1130～1170	表示GAMEOVER
1300～1350	用戶定義文字
1500～1670	說明遊戲中的圖象數值



<照片4>進入下一級



<照片5>遊戲結束GAME O－VER

```
10 DIM M(25,35):DIM N(25,35):DIM
 L(25,35):DIM P(25,35)
15 DIM Y(6):DIM X(6)
16 DIMD(19):DIMR(10,30)
40 GOSUB1500
50 GOSUB1300
120 PRINT
135 PRINT
140 PRINT TAB(11);"■ ──→ 10 POIN
TS"
145 PRINT
150 PRINT TAB(11);"ENARGY ──→ BO
NUS"
160 PRINT
170 PRINT TAB(14);"  ↑"
180 PRINT TAB(14);"  U"
190 PRINT TAB(14);"←H J→"
200 PRINT TAB(14);"  N "
210 PRINT TAB(14);"  ↓"
220 PRINT
230 PRINT TAB(14);"☺ ☺ ☺"
240 PRINT
250 PRINTTAB(11);"HIT ANY KEY"
270 PICK Q
280 IF Q=0 GOTO 270
290 GOTO 300
295 REM*********************
300 CLS
305 S=0:W=6:Z=3:W1=1:P1=0
310 LOCATE 10,10:PRINT"WAIT A MI
NUTE"
320 W=W-1
330 IF W=0 THEN W=5:W1=W1+1
340 FOR T=1 TO W
350 Y(T)=RND(13)+5:X(T)=RND(13)+
4
355 IF Y(T)=10 THEN IF X(T)=12 G
OTO 350
360 NEXT T
370 S3=0:E=600
380 A=10:B=12
390 FOR P=1 TO 25
400 FOR Q=1 TO 35
410 M(P,Q)=1:N(P,Q)=1:P(P,Q)=0
420 NEXT Q
430 NEXT P
440 FOR P=4 TO 18
450 FOR Q=3 TO 17
460 P(P,Q)=1:L(P,Q)=1
470 NEXT Q
480 NEXT P
500 CLS
540 PRINT
550 PRINT
560 PRINT
570 FOR T=1 TO 17
580 PRINT"  ■                    ■  ■
 ■"
590 NEXT T
600 LOCATE 3,2:PRINT"████████████
███████  ███████████"
610 LOCATE 19,2:PRINT"███████████
████████   ███████████"
620 LOCATE 10,20:PRINT"
"
630 LOCATE 11,20:PRINT"
"
640 LOCATE9,22:PRINT"████████":LO
CATE12,22:PRINT"████████"
660 LOCATE 5,23:PRINT"SCORE"
670 LOCATE 14,23:PRINT"ENE"
680 LOCATE 15,25:PRINT"RGY"
690 REM
700 REM
705 OPTION CMODE1
710 FOR T=1 TO W
720 M1=Y(T):M2=X(T)
730 M(M1,M2)=1
740 V=RND(W1)+1
750 IF Y(T)>A THEN Y(T)=Y(T)-V
760 IF Y(T)<A THEN Y(T)=Y(T)+V
770 IF X(T)>B THEN X(T)=X(T)-V
780 IF X(T)<B THEN X(T)=X(T)+V
790 M3=Y(T):M4=X(T)
800 IF P(M3,M4)=0 THEN Y(T)=M1:X
(T)=M2
810 M5=Y(T):M6=X(T):M(M5,M6)=0
850 LOCATE M5,M6:PRINTFLD(1);" "
851 D$=" "
852 IF N(M1,M2)=-1 THEN D$="■"
853 LOCATE M1,M2:PRINTFLD(0);D$
855 IF(M5=A)*(M6=B) GOTO 1070
860 FOR T1=1 TO 3
870 M7=A:M8=B
880 PICK G
890 IF G=$55 THEN A=A-1:X$="┬":G
OTO935
900 IF G=$4E THEN A=A+1:X$="┴":G
OTO935
910 IF G=$48 THEN B=B-1:X$="├":G
OTO935
920 IF G=$4A THEN B=B+1:X$="┤":G
OTO935
930 GOTO 940
935 LOCATE M7,M8:PRINT"■"
940 IF P(A,B)<1 THEN A=M7:B=M8
950 LOCATE A,B:PRINTX$
960 IF M(A,B)=0 GOTO 1070
970 N(A,B)=-1:P(A,B)=-1
980 S3=S3+L(A,B)
990 IF S3=200 GOTO 1180
1000 E=E-1
1010 IF E=0 GOTO 1070
1015 S=S+L(A,B):L(A,B)=0
1020 LOCATE 7,23:PRINTS;"0 "
1030 LOCATE 17,25:PRINT E;" "
1040 NEXT T1
1050 NEXT T
1060 GOTO 710
1070 Z=Z-1
1080 IF Z=0 GOTO 1130
1090 LOCATE 10,5:PRINT"LEFT ";Z
1100 REM
1110 REM
1120 GOTO 340
1130 LOCATE 10,4:PRINT" GAME OVE
R ":FORI=0TO5000:NEXTI
1133 PICKO
1134 IFO<>0THENGOTO1136
1135 GOTO1133
1136 GOTO295
1140 END
1150 IF Q=0 GOTO 1140
1160 GOTO 1170
1170 GOTO 20
1180 P1=P1+1
1190 LOCATE 10,4:PRINTP1;" P. CL
EAR!"
1200 S=S+E
1210 LOCATE 7,23:PRINT S;"0 "
1220 GOTO 320
1300 POKE$C000,$3C:POKE$C001,$7F
1310 POKE$C002,$FF:POKE$C003,$49
1320 POKE$C004,$FF:POKE$C005,$FF
1330 POKE$C006,$FF:POKE$C007,$81
1350 RET
1500 CLS
1510 PRINT:PRINT:PRINT
1520 RESTORE
1530 FORT1=3TO7
1540 FORT=2TO29
1550 READR(T1,T)
1560 LOCATET1,T:PRINT"┤"
1570 R$="■"
1580 IFR(T1,T)=0THENR$=" "
1590 LOCATET1,T:PRINTR$
1600 NEXTT
1610 NEXTT1
1620 RET
1630 DATA1,1,1,1,1,0,1,1,1,1,1,0
,1,1,1,0,1,0,0,0,1,0,1,1,1,1,1,0
1640 DATA1,0,0,0,1,0,1,0,0,0,1,0
,0,1,0,0,1,1,0,0,1,0,0,0,1,0,0,0
1650 DATA1,1,1,1,1,0,1,1,1,1,1,0
,0,1,0,0,1,0,1,0,1,0,0,0,1,0,0,0
1660 DATA1,0,0,0,0,0,1,0,0,0,1,0
,0,1,0,0,1,0,0,1,1,0,0,0,1,0,0,0
1670 DATA1,0,0,0,0,0,1,0,0,0,1,0
,1,1,1,0,1,0,0,0,1,0,0,0,1,0,0,1
```

本刊投稿規例：

爲方便大家發表各電腦程式作品，並減輕本刊編輯之工作。現特定各投稿之格式。

1) 解說用之內文需分爲下列各數。

程序計算機：500字以上。

袋 裝 電 腦：1000字以上。

各微型電腦：1500字以上。

2) 各投稿程式，需用印字機印出，如未能者可用打字機代之。

3) 需要付上其錄音帶或磁碟之版本，方便測驗該程式是否有錯。

4) 如程式是轉載於其他書籍者，須明確指出該書籍的名稱及期數。

《CASIO FP-1000 FP-1100介紹》

金石是日本有名生產計算機公司，自從在198
0年，正式推出FX—9000P枱型微型電腦後，
已能把握到生產微型電腦之技術，在今年年尾將會推
出FP—1000及FP—1100微型電腦，而兩
系統在基本結構上，售賣價錢大約二千至三千元左右
，這售價是日本的售買價，相信香港售價和日本相差
不遠。

這兩系統大致上結構是相同的，而它們所採用的
微型處理器爲Z—80A，速度爲4MHZ，內部除
了Z—80A外，還有另外一粒8BIT的微型處理
器用來附起輸入和輸出之控制，這樣可以使到電腦的
本身運算不會因其它的輸入輸出，產生片斷（INT
—ERRUPT）而速度減慢。—

至於，主體記憶容量方面，FP—1100本身
已有64K RAM，還可以擴展加多48K RA
—M，而FP—1000主體記憶容量比FP—11
00小，本身已有16K，可擴展加多48K，至F
P—1100系統中其RAM的容量合共112K
RAM。ROM方面兩系統ROM的容量相同，爲3
6K ROM，有32K BASIC的編譯器，4
K爲輸入輸出的控制，同時此機的結構已違反電腦上
的結構概念，因爲機本身RAM可以加到240K至
ROM最大可以擴展到292K，因爲在8BIT的
系統中最大的主體記憶爲128K RAM，而FP
—1100可以擴展到那麼大，原因此系統同時使用
二粒微處理器。

金石兩系統的語言名爲C82—BASIC，是
完善的軟件語言，程式的行數最大可以到6499，
在單精密度有效值爲6個位，倍精密度爲16個位，
但另外比其它電腦多一項的精密度，其位值可到24
個位。指數最大可由+99到最小−99，每字元最
長可達255個，機本身可以自行計算八進及十六進
之數字，對於在程序上的運用非常方便，在BAS—
IC語言的命令和NEC8801非常相似。

FP—1100及FP—1000畫面最示方面
，可以顯示每行80個字共25行，及每行40字可
以20行或25行，在FP—1100可以連按日本
漢字ROM可顯示日本中文，和中國中文大同小異，
圖按方面，在FP—1000其繪圖的畫面由640
×200點組成，但只可顯示黑白，而FP—1100
0最大可以640×400，其次爲640×200
有三個畫面（PAGE），再其次爲640×200
一個畫面（PAGE）有八種顯示。因能顯示640
×200點，顏色可以互相混合產生另一種顏色，輸
出方面採用RG. B系統或連接綠色顯示器及N. T
S. C電視機。

機的本身可以連接錄音機其輸出及輸入速度由3
00至1200BAUD，由用者去選擇，同時有二
個可以擴展座可以連磁碟等，更可印字輸出裝置，外
面設計和N. E. C. 8801外型差不多，鍵盤的
設計對於程式設計的用家非常喜愛，因爲鍵字有95
個，而功能鍵有10個。

評價CASIO FP—1000及FP—11
00是好的電腦，但只可嘆是推出市場販買時間太慢
，是否佔一席位呢，因日本已有多間公司生產16B
—IT的私人電腦，販買價大約6千港元左右。

IE 80

高樓大廈中的遊戲

32K

NEC 6001



在高樓林立的大街上，由於大地震，使得高樓的中間屋頂倒塌，這時樓上的會跳躍的人，就像是比賽氣球的遊戲一樣，由上向下跳躍。這是一個非常滑稽的遊戲，大家玩時均會感到很開心。另外，這遊戲要使用ＲＯＭ／ＲＡＭ的插板。

有關程序

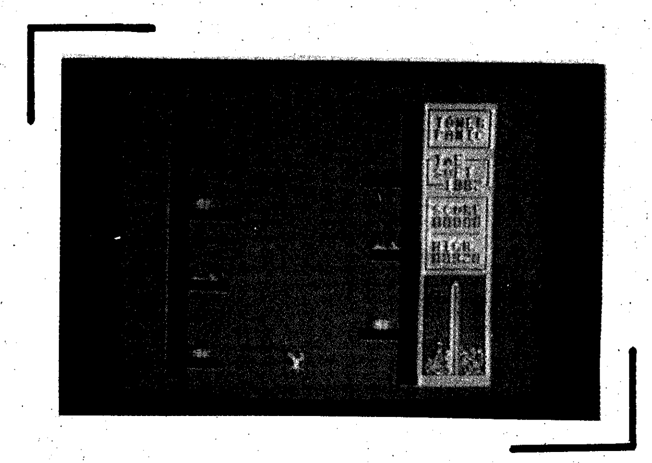
會跳躍的人在跳躍之際，若不能上下移動，就會在屏幕上看到一些梳狀放射性的線條，好像在飛濺一般。起初爲＋３點，隨後爲＋２，＋１，０，－１，－２，－３等的變動，對於落下物體加上這些數值，總的使整個畫面的運動感到很柔和自然。

另外，在ＰＣ－６００１的彩色圖像上（４色），由於橫向的寬度寬一些（即爲縱向的二倍），使得在橫向的文字數不足。爲此，應出８×８ｂｉｔ（實際８×４ｂｉｔ）的文字數據和顯示程序。這樣使用時，應把文字數據放到ＢＤ３ＣＨ～ＢＦＦ３Ｈ之中，把顯示的子程序放到ＢＣＥ４Ｈ～ＢＤＩＢＨ之中。

輸入程序

在輸入程序之前，請把頁面寄存器指定爲１，這個程序可由機器語言和ＢＡＳＩＣ寫成的。

首先，輸入ＢＡＳＩＣ，請保存這個程序。接著把機器語言（ＡＢ６０Ｈ～ＢＥＦＦＨ）輸入到Ｉ／Ｏ監視器中，並像ＢＡＳＩＣ程序那樣地保存。（把監視器的１０行的ＣＬＥＡＲ４００，＆ＨＤ８００的Ｄ８００作爲ＡＢ５Ｆ，以輸入機器語言）。



程序的起動

把頁數指定爲 4 ，並帶上ＢＡＳＩＣ程序。運行（ＲＵＮ）程序，請指定短路鍵。（Ｊ　ＯＲ　Ｋ鍵），接著轉向機器語言，其後就可開始表演了。

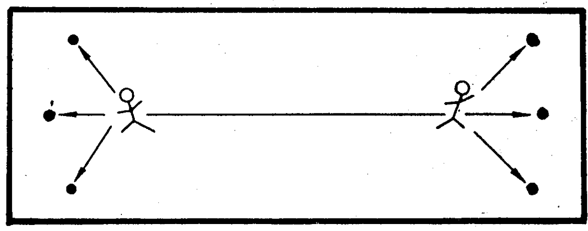
遊戲的玩法

按下ＳＰＡＣＥ鍵，遊戲放置起點了。這時高樓大廈在振動中，跳躍的人在落下，此遊戲還不能開始玩。

在圖像的右邊是得分表及高得分表，其下表示高樓大廈的全景（即爲美麗的高層建築夜景），用閃光點來表示現在的位置，其次是地震幸存者所剩下人的閃光點。

• 人的移動

跳躍的人發出放射線在跳動著，而由短路鍵或指針鍵，可控制他向 8 個方向移動。但是不能向前進和後退。若他已落到地上，就還可以慢慢向上跳躍，若碰到左右牆壁或台面側面時，會再退返回來。另外，當他碰到落下物體時，仍然可返回，只有落下物正好打到他 的頭上，他就會被〝打死〞。



• 錢袋和障礙物

當跳躍的人正好跳到錢袋上時，可加６０分，若跳到障礙物上，就算〝死了〞。

• 投石頭

按下觸發或ＳＰＡＣＥ鍵，跳躍的人就會投出石子，但方向只能限於向上傾斜或向下傾斜。若短路開關或指針鍵不是向上或向下，就不能投出石子。總之它能向上或向下斜拋出石子。

• 落下物

在遊戲中，會有各種各樣的東西落下來。這個跳躍的也可以投石頭，當它打中妖精時可得分，若石頭碰到自己的頭爲〝死亡〞。在表演時可計算出各種得分。

若用障礙物擋住石子，就能把石子消去，若用錢袋去頂住，則錢袋消失，沒有得分。另外，投出石子向什麽方向時，從圖右到輸出點，方向在不斷變化，當跳躍的人碰到它時，就會被打死。當圖面內有石頭時，第一塊石頭就不會再落下。

• 其他

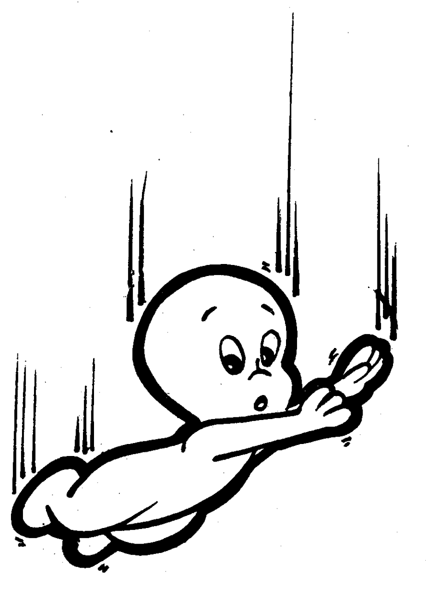
屏幕上的人不跳躍時，就會慢慢落下，這時跳躍的人就爲〝死了〞。大廈全景圖爲 2 點。若你登上一次這樣的高樓大廈，就可得５０００分。

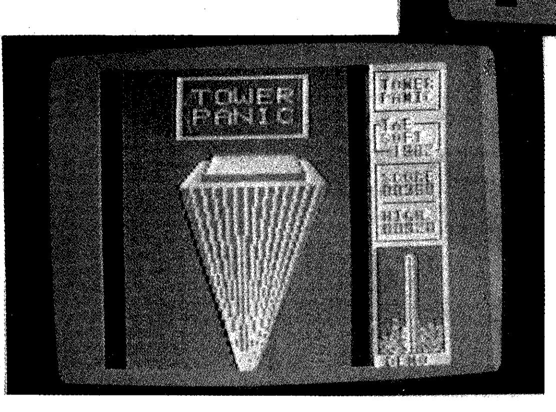
可增加一個跳躍的人，若跳躍的人死了，它就會在原來位置產生另一個人。並使現有狀態不變。

• 結語

正如說明中已說過的一樣作爲一個遊戲，內容應該是可以理解的。要得到高的積分，就要正確地進行操作。

設置的障礙物也是如此，當障礙物碰到石頭時，它會自動被消除，而且要掌握移動的竅門。





表一：ＢＡＳＩＣ程序表

```
10 REM***********************
20 REM*    TOWER PANIC      *
30 REM*    For PC-6001      *
40 REM*                     *
50 REM*                     *
60 REM***********************
70 SCREEN1,1,1:COLOR1,1,1:CLS:CONSOLE,,0,0
80 PRINT"KEY or JOYSTICK ?":PRINT:PRINT"PUSH (K) or (J) KEY"
90 K$=INKEY$:R=RND(1):IFK$="j"THENKJ=1:GOTO120
100 IFK$="k"THENKJ=0:GOTO120
110 GOTO90
120 GOSUB1520:EXEC&HF87C
130 SCREEN3,2,2:SCREEN3,3,3
140 P0$="S13M300T255L8V1503g04ceR20gR10eL1g"
150 P2$="S13M300T255L8V1505b06egR20bR10gL1b"
160 P3$="S13M300T255L6V1506gR607cL406b"
170 P4$="L5aR35gR64fedR64cR407cR50L4cL2R64c"
180 D=0
190 GOSUB1610:EXEC&HB6E0
200 GOSUB750
210 FORI=0TO90:ST=STRIG(0):IFST=1THEND=1
220 NEXTI:IFD=1THEN250
230 GOSUB1060:GOSUB1440
240 IFPP=1THEN550
250 REM** PLAY **
260 POKE&HBF65,&H2F
270 S1=1:M1=2:POKE&HBF62,0:POKE&HBF63,0:POKE&HBF64,0:GOSUB1430
280 GOSUB1510:GOSUB1610
290 POKE&HBF54,S1:POKE&HBF55,KJ:POKE&HBF5C,M1
300 SCREEN,,3:EXEC&HB6E0
310 EXEC&HB964:EXEC&HBAC5:EXEC&HB36D:GOSUB1460:GOSUB1210
320 R=INT(RND(1)*256):POKE&HBF04,R:FORI=0TO590:NEXTI:M1=M1-1:GOSUB1460
330 POKE&HBF5B,1:EXEC&HAD00
340 P1=PEEK(&HBF5D):IFP1=1THEN420
350 IFS1=1THENM1=M1+1
360 M1=M1+1:GOSUB1410:IFSC=2THENSC=3:GOTO380
370 SC=2
380 S1=S1+1:POKE&HBCFD,0:RESTORE1800:O=0:GOSUB1310:GOSUB1310
390 GOSUB1310
400 POKE&HBF64,0:POKE&HBF65,&H2F:POKE&HBCFD,&HFF:EXEC&HAB60
410 FORI=0TO150:NEXTI:GOTO280
420 GOSUB1410:IFSC=2THENSC=3:GOTO440
430 SC=2
440 EXEC&HB36D:COLOR,,2:GOSUB1440
450 GOSUB1510:COLOR,,1:EXEC&HAB70:GOSUB1440
460 IFM1<0THEN480
470 GOTO280
480 GOSUB1410:IFSC=3THENSC=2:GOTO500
490 SC=3
500 SCREEN,SC,SC:LOCATE4,3:PRINT"GAME"
510 PRINTSPC(4)"OVER":D=0
520 I=100
530 Y1=STRIG(0):IFY1=1GOTO250
540 I=I-1:IFI<>0GOTO530
550 POKE&HBCFD,0:PH=&HBF66:REM** DEMO 2 **
560 O=1:RESTORE1710:GOSUB1290:IFV=1GOTO730
570 GOSUB1320:IFV=1THEN730
580 POKEPH,&H28:POKEPH+1,&H40:POKEPH+2,&HFC
590 POKEPH+3,&HBA:GOSUB1440:EXEC&HAC73
600 GOSUB1320:IFV=1THEN730
610 POKEPH+1,&H58:POKEPH+2,&HC4:POKEPH+3,&HB4:GOSUB1440:EXEC&HAC73
620 GOSUB1320:IFV=1THEN730
630 POKEPH+1,&H70:POKEPH+2,&H08:POKEPH+3,&HB5:GOSUB1440:EXEC&HAC73
640 GOSUB1320:IFV=1THEN730
650 POKEPH+1,&H88:POKEPH+2,&H4C:GOSUB1440:EXEC&HAC73
660 GOSUB1320:IFV=1THEN730
670 POKEPH+1,&HA0:POKEPH+2,&HB2:GOSUB1440:EXEC&HAC73
680 GOSUB1320:IFV=1THEN730
690 I=50:POKE&HBCFD,&HFF
700 S3=STRIG(0):IFS3=1GOTO250
710 I=I-1:IFI=0GOTO200
720 GOSUB1430:GOTO700
730 POKE&HBCFD,&HFF:GOTO250
740 POKE&HBCFD,&HFF
750 REM** DEMO 1 **
760 GOSUB1410
770 FORI=0TO1:SCREEN3,I+2,SC
780 LINE(16,0)-(183,191),3,BF
790 GOSUB1020
800 LINE(56+I*16,56)-(135+I*16,56),2
810 LINE(40+I*16,71)-(151+I*16,71),2
820 LINE(56+I*16,56)-(40+I*16,71),2
830 LINE(135+I*16,56)-(151+I*16,71),2
840 LINE(96,191)-(109,191),2
850 LINE(40+I*16,71)-(96,191),1
860 LINE(151+I*16,71)-(109,191),1
870 PAINT(104,64),2,2
880 LINE(40+I*16,72)-(151+I*16,72),1
890 LINE(96,190)-(109,190),1
900 PAINT(104,104),1,1
910 LINE(56+I*16,68)-(135+I*16,64),1,BF
920 LINE(64+I*16,52)-(127+I*16,52),1
930 LINE(56+I*16,64)-(64+I*16,52),1
940 LINE(135+I*16,64)-(127+I*16,52),1
950 PAINT(104,56),2,1
960 FORJ=0TO12
970 LINE(48+J*8+I*16,72)-(96+J,190),2:NEXTJ:NEXTI
980 FORI=0TO9:FORJ=2TO3
990 FORC=1TO15:NEXTC:SCREEN,,J
1000 ST=STRIG(0):IFST=1THEND=1
1010 GOSUB1250:FORC=1TO10:NEXTC:NEXTJ:NEXTI:RETURN
1020 LINE(16,0)-(183,191),3,BF
1030 LOCATE3,0:COLOR2,3,1:PRINT""
1040 PRINTSPC(3)"TOWER":PRINTSPC(3)"PANIC"
1050 PRINTSPC(3)"":RETURN
1060 REM** DEMO PLAY **
1070 GOSUB1610:GOSUB1430:POKE&HBF5B,0:POKE&HBF62,0:POKE&HBF63,0
1080 POKE&HBF64,0
1090 POKE&HBF65,&H2F:GOSUB1410:IFSC=3THENSC=2:GOTO1110
1100 SC=3
1110 SCREEN,,SC:POKE&HACE1,&H30
1120 GOSUB1280:GOSUB1210:POKE&HBF6C,23:RESTORE1700:FORI=0TO3
1130 READR:POKE&HBF6A,R:POKE&HBF6B,26+I:EXEC&HAC7E:NEXTI
1140 FORI=0TO590:NEXTI
1150 R=INT(RND(1)*256):POKE&HBF04,R
1160 EXEC&HAD00:PP=PEEK(&HBF5D):IFPP=0THENRETURN
1170 GOSUB1410:IFSC=2THENSC=3:GOTO1190
1180 SC=2
1190 EXEC&HB36D:COLOR,,2:GOSUB1440:GOSUB1510:COLOR,,1
1200 EXEC&HAB70:RETURN
1210 GOSUB1430
1220 PLAYP0$,P0$,P0$
1230 PLAYP2$,P2$,P0$
1240 PLAYP3$:PLAYP4$:RETURN
1250 SOUND0,60:SOUND1,9:SOUND6,2:SOUND7,7
1260 SOUND8,16:SOUND9,16:SOUND10,16
1270 SOUND12,220:SOUND13,0:RETURN
1280 EXEC&HB6E0:EXEC&HB964:EXEC&HBAC5:EXEC&HB36D:RETURN
1290 GOSUB1410:GOSUB1510:IFSC=2THENSC=3:GOTO1310
1300 SC=2
1310 SCREEN,SC,SC
1320 READW:READY:READX:POKE&HBF6C,Y:W1=X+W:V=0
1330 POKE&HBF6B,X:READE:POKE&HBF6A,E:EXEC&HAC7E
1340 SOUND0,70:SOUND1,0:SOUND7,&HFE:SOUND8,15:SOUND11,253
1350 SOUND12,23:SOUND13,1
1360 GOSUB1430
1370 IFW1=XTHENRETURN
1380 IFO=0THEN1400
1390 ST=STRIG(0):IFST=1THENV=1:RETURN
1400 X=X+1:GOTO1330
1410 P=PEEK(&HBF01):IFP=&HC2THENSC=2:RETURN
1420 SC=3:RETURN:REM** NOW SCREEN **
1430 SOUND7,250:SOUND8,64:SOUND9,64:SOUND10,64:RETURN
1440 SOUND0,200:SOUND2,20:SOUND6,15:SOUND7,&HCC:SOUND8,16
1450 SOUND9,16:SOUND12,50:SOUND13,9:RETURN
1460 POKE&HBF6A,&H24:POKE&HBF6C,23:FORI=26TO29
1470 POKE&HBF6B,I:EXEC&HAC7E:NEXTI
1480 IFM1<0THENRETURN
1490 POKE&HBF6A,&H34:X=26:FORI=0TOM1
1500 POKE&HBF6B,X+I:EXEC&HAC7E:NEXTI:RETURN
1510 FORI=2TO3:SCREEN,I,SC:LINE(16,0)-(183,191),3,BF:NEXTI:RETURN
1520 RESTORE1550:FORI=&HF878TO&HF8D1
1530 READA$:POKEI,VAL("&h"+A$)
1540 NEXTI:RETURN
1550 DATA 1a,c3,06,1b,cd,9a,25,06,05,cd,70,1a,fe,af,20,f7
1560 DATA 10,f7,21,32,f8,06,06,cd,70,1a,77,23,10,f9,cd,06
1570 DATA 1b,af,32,38,f8,21,76,25,cd,cf,30,21,36,f8,cd,cf
1580 DATA 30,cd,9a,25,2a,32,f8,ed,4b,34,f8,1e,00,cd,70,1a
1590 DATA 57,83,5f,72,23,0b,78,b1,20,f3,cd,70,1a,bb,ca,aa
1600 DATA 1a,cd,06,1b,21,85,03,c3,cf,30
1610 RESTORE1640:FORI=&HBF00TO&HBF5F
1620 READA$:POKEI,VAL("&h"+A$)
1630 NEXTI:RETURN
1640 DATA AA,C2,DA,C0,FF,00,18,10,01,50,10,00,80,10,02,B0
1650 DATA 10,01,E0,10,01,00,98,00,38,A0,02,68,90,01,98,98
1660 DATA 00,C8,A0,00,10,10,01,48,10,02,78,10,00,A8,10,02
1670 DATA D8,10,02,F8,A0,00,30,A0,01,60,A0,02,90,A0,01,C0
1680 DATA A0,02,78,1A,03,FE,00,00,00,06,06,07,00,40,07,00
1690 DATA 60,07,00,78,01,01,00,00,00,00,00,00,02,00,FF,00
1700 DATA &H0D,&H0E,&H16,&H18
1710 DATA 16,2,4,&H2C,&H2C,&H24,&H1D,&H18,&H20,&H0E,&H1B,&H24
1720 DATA &H19,&H0A,&H17,&H12,&H0C,&H24,&H2D,&H2D
1730 DATA 16,4,4,&H1D,&H25,&H0E,&H24,&H1C,&H18,&H0F,&H1D,&H24
1740 DATA &H19,&H1B,&H0E,&H1C,&H0E,&H17,&H1D,&H1C
1750 DATA 10,7,9,&H29,&H24,&H24,6,0,&H24,&H19,&H18,&H12,&H17,&H1D
1760 DATA 10,10,9,&H29,&H24,1,0,0,&H24,&H19,&H18,&H12,&H17,&H1D
1770 DATA 10,13,9,&H29,&H24,2,0,0,&H24,&H19,&H18,&H12,&H17,&H1D
1780 DATA 10,16,9,&H29,&H24,3,0,0,&H24,&H19,&H18,&H12,&H17,&H1D
1790 DATA 10,19,9,&H29,&H24,4,0,0,&H24,&H19,&H18,&H12,&H17,&H1D
1800 DATA 8,3,8,&H1F,&H0E,&H1B,&H22,&H24,&H10,&H18,&H18,&H0D
1810 DATA 11,6,7,&H0B,&H18,&H17,&H1E,&H1C,&H24
1820 DATA &H2C,5,0,0,0,&H2D
1830 DATA 16,9,4,&H22,&H18,&H1E,&H24,&H1D,&H1B,&H22,&H24
1840 DATA &H17,&H0E,&H20,&H24,&H1D,&H18,&H20,&H0E,&H1B
```

```
AB60:2A 62 BF 01 F4 01 09 22:6C
AB68:62 BF C9 00 00 00 00 00:EA
AB70:3E 0A F5 21 00 BF CB 0E:F6
AB78:DA 87 AB 23 36 C2 23 36:80
AB80:DA 23 36 C0 C3 90 AB 23:14
AB88:36 E2 23 36 FA 23 36 E0:A4
AB90:CD 3A B6 CD 4C B6 CD 5E:B7
AB98:B6 CD 88 AC F1 A7 C6 02:17
ABA0:DB F5 32 E1 AC FE D0 D2:2C
ABA8:B3 AB CD FF AB CD DF AB:2C
ABB0:C3 BB AB 3A 42 BF C6 04:2E
ABB8:32 42 BF CD 64 B9 CD C5:AF
ABC0:BA CD 12 AC 3A 03 BF FE:3F
ABC8:C0 CA D1 AB 3E 01 C3 D3:DB
ABD0:AB 3E 02 CD ED 13 21 DF:B8
ABD8:AC CD 1C BD C3 73 AB 06:39
ABE0:13 78 FD 21 05 BF 1E 03:8E
ABE8:CD 60 BB FD 35 01 FD 35:4D
ABF0:01 FD 35 01 FD 35 01 05:6C
ABF8:78 FE FF C2 E1 AB C9 3A:C6
AC00:43 BF FE 5C CA 0E AC D2:B2
AC08:0C AC C6 04 D6 02 32 43:CF
AC10:BF C9 3A 00 BF E6 01 C2:2A
AC18:20 AC 11 2F AC C3 23 AC:4A
AC20:11 51 AC 3A 42 BF 67 3A:EA
AC28:43 BF 6F CD 6A BB C9 02:2E
AC30:10 80 20 B0 2C B0 0C 0C:54
AC38:0C 0F 3C 03 F0 03 C0 03:10
AC40:C0 03 C0 03 C0 03 C0 0F:18
AC48:F0 32 8C 82 82 82 82 02:B8
AC50:80 02 10 08 02 3B 0E 30:12
AC58:0E 30 30 3C F0 0F C0 03:6C
AC60:C0 03 C0 03 C0 03 C0 83:8C
AC68:C2 BF FE 02 80 02 80 02:85
AC70:80 02 80 2A 66 BF ED 5B:99
AC78:68 BF CD E8 BB C9 3A 6A:04
AC80:BF 2A 6B BF CD E4 BC C9:49
AC88:CD 83 B6 C3 21 BB FF FF:A0
AC90:3A E1 AC 3C 3C FE FA DA:11
AC98:9C AC 3E FA 47 3A 41 BF:01
ACA0:FE 02 C2 A7 AC 06 1E 78:B1
ACA8:32 E1 AC 06 02 78 FD 21:5D
ACB0:4B BF 1E 03 CD 60 BB FD:10
ACB8:7E 00 FE 04 CA D1 AC FE:C5
ACC0:05 CA D1 AC 05 78 FE FF:C6
ACC8:C2 AD AC 21 DF AC C3 D4:5E
ACD0:AC 21 EA AC E5 CD 2C BD:FE
ACD8:E1 CD 1C BD C3 00 AD 05:FC
ACE0:00 1E 07 FE 08 10 0C 96:DD
ACE8:0D 0A 06 00 64 06 05 07:93
ACF0:F6 08 10 0C 14 0D 09 00:44
ACF8:00 00 00 00 00 00 00 00:00
AD00:AF 32 28 FA 21 00 BF CB:AE
```

```
AD08:0E DA 18 AD 23 36 C2 23:EB
AD10:36 DA 23 36 C0 C3 21 AD:BA
AD18:23 36 E2 23 36 FA 23 36:E7
AD20:E0 CD 3A B6 CD 4C B6 CD:39
AD28:5E B6 CD 83 B6 3A 5B BF:6E
AD30:A7 CA 85 AD 3A 55 BF A7:98
AD38:C2 5F AD CD 61 10 47 E6:39
AD40:04 32 56 BF 78 E6 08 32:E3
AD48:57 BF 78 E6 20 32 58 BF:DD
AD50:78 E6 10 32 59 BF 78 E6:16
AD58:80 32 5A BF C3 AB AD 3E:24
AD60:01 CD A6 1C 47 E6 01 32:F0
AD68:56 BF 78 E6 02 32 57 BF:BD
AD70:78 E6 04 32 58 BF 78 E6:09
AD78:08 32 59 BF 78 E6 10 32:F2
AD80:5A BF C3 AB AD CD 61 10:72
AD88:E6 80 C0 CD 9F BC FE 10:5C
AD90:D2 A7 AD 3E 01 32 5A BF:B0
AD98:21 56 BF CD 9F BC E6 01:45
ADA0:85 6F 3E 01 C3 AB AD AF:F5
ADA8:32 5A BF 3A 46 BF A7 CA:FB
ADB0:DE AD 3A 47 BF 4F 3A 42:96
ADB8:BF D6 02 B9 D2 DE AD C6:73
ADC0:10 B9 DA DE AD 3A 48 BF:6F
ADC8:4F 3A 43 BF D6 04 B9 D2:F0
ADD0:DE AD C6 10 B9 DA DE AD:7F
ADD8:3E 01 32 5D BF C9 06 02:5E
ADE0:78 FD 21 4B BF 1E 03 CD:8E
ADE8:60 BB FD 7E 00 FE 07 CA:65
ADF0:50 AE FE 04 DA FE AD 3C:C1
ADF8:FD 77 00 C3 50 AE 3A 46:B5
AE00:BF A7 CA 50 AE 3A 47 BF:6E
AE08:4F FD 7E 01 D6 02 B9 D2:2E
AE10:50 AE C6 10 B9 DA 50 AE:65
AE18:3A 48 BF 4F FD 7E 02 D6:E3
AE20:04 B9 D2 50 AE C6 10 B9:1C
AE28:DA 50 AE 21 0A 00 11 0A:1E
AE30:00 FD 7E 00 A7 CA 3D AE:D7
AE38:19 3D C3 34 AE ED 5B 62:A5
AE40:BF 19 22 62 BF FD 36 00:4E
AE48:04 AF 32 46 BF C3 57 AE:B2
AE50:05 78 FE FF C2 E0 AD 06:CF
AE58:02 78 FD 21 4B BF 1E 03:C3
AE60:CD 60 BB FD 7E 00 FE 07:68
AE68:C2 9A AE C5 CD 9F BC C1:B8
AE70:4F 3A 54 BF 87 B9 DA 9A:50
AE78:AE C5 CD 9F BC C1 E6 03:45
AE80:FE 03 C2 93 AE 3A 54 BF:51
AE88:FE 02 D2 91 AE AF C3 93:16
AE90:AE 3E 03 FD 77 00 FD 36:96
AE98:01 00 05 78 FE FF C2 59:96
AEA0:AE 3A 46 BF A7 CA F7 AE:03
AEA8:06 09 78 FD 21 23 BF 1E:A5
```

```
AEB0:03 CD 60 BB FD 7E 00 FE:64
AEB8:02 CA F0 AE 3A 47 BF 4F:F9
AEC0:FD 7E 01 C6 04 B9 D2 F0:C1
AEC8:AE C6 0C B9 DA F0 AE 3A:EB
AED0:48 BF 4F FD 7E 02 D6 04:AD
AED8:B9 D2 F0 AE C6 18 B9 DA:9A
AEE0:F0 AE FD 36 00 02 AF 32:B4
AEE8:46 BF CD 95 B6 C3 F7 AE:85
AEF0:05 78 FE FF C2 AA AE 06:9A
AEF8:09 78 FD 21 23 BF 1E 03:A2
AF00:CD 60 BB FD 7E 00 FE 02:63
AF08:CA 4C AF FD 7E 01 FE 81:C0
AF10:D2 4C AF FE 78 DA 4C AF:18
AF18:3A 43 BF 4F FD 7E 02 D6:DE
AF20:0C B9 D2 4C AF C6 20 B9:31
AF28:DA 4C AF FD 7E 00 A7 C2:B9
AF30:46 AF 11 06 00 2A 62 BF:57
AF38:19 22 62 BF FD 36 00 02:91
AF40:CD 95 B6 C3 53 AF 3E 01:1C
AF48:32 5D BF C9 05 78 FE FF:91
AF50:C2 F9 AE 3A 43 BF FE 38:DB
AF58:DA 6E AF FE 80 D2 6E AF:64
AF60:3A 41 BF FE 03 C2 BE AF:1A
AF68:3E 01 32 5D BF C9 3A 41:D1
AF70:BF FE 02 DA AA AF C2 8E:42
AF78:AF 3E FF 32 5E BF AF 32:1C
AF80:5F BF 3A 45 BF EE FF 3C:85
AF88:32 45 BF C3 AA AF FE 03:53
AF90:C2 9B AF 3E 02 32 5F BF:9C
AF98:C3 AA AF FE 04 C2 A5 AF:34
AFA0:3E FE C3 A7 AF 3E 02 32:C7
AFA8:45 BF 3A 5F BF FE 02 CA:26
AFB0:E4 AF A7 C2 D1 AF 3A 5E:14
AFB8:BF D6 08 DA C4 AF 32 5E:7A
AFC0:BF C3 E4 AF 3E 40 32 5E:23
AFC8:BF 3E 01 32 5F BF C3 E4:F5
AFD0:AF 3A 5E BF C6 08 DA DF:8D
AFD8:AF 32 5E BF C3 E4 AF 3E:92
AFE0:02 32 5F BF 3A 5E BF 07:B0
AFE8:07 E6 03 47 3A 5F BF A7:36
AFF0:C2 FA AF 78 32 44 BF C3:DB
AFF8:0E B0 FE 01 C2 09 B0 78:B0
B000:EE FF 3C 32 44 BF C3 0E:2F
B008:B0 3E FD 32 44 BF 3A 45:9F
B010:BF FE 02 C2 34 B0 47 3A:E6
B018:59 BF A7 CA 20 B0 04 04:61
B020:3A 58 BF A7 CA 4F B0 3A:FB
B028:00 BF E6 01 CA 4F B0 05:74
B030:05 C3 4F B0 47 3A 58 BF:5F
B038:A7 CA 3E B0 05 05 3A 59:FC
B040:BF A7 CA 4F B0 3A 00 BF:28
B048:E6 01 CA 4F B0 04 04 3A:F2
B050:43 BF 80 32 43 BF 21 44:1B
```

```
B058:BF 3A 56 BF A7 CA 61 B0:90
B060:34 3A 57 BF A7 CA 69 B0:0E
B068:35 3A 46 BF A7 CA 9C B0:31
B070:3A 4A BF 47 3A 48 BF 80:4B
B078:32 48 BF 3A 49 BF 47 3A:FC
B080:47 BF 80 47 3A 44 BF 80:8A
B088:FE 0B DA 98 B0 FE D0 D2:CB
B090:98 B0 32 47 BF C3 F9 B0:EC
B098:AF 32 46 BF 3A 5A BF A7:E0
B0A0:CA F9 B0 3A 56 BF A7 C2:2B
B0A8:B4 B0 3A 57 BF A7 C2 B4:D1
B0B0:B0 C3 F9 B0 FD 21 46 BF:3F
B0B8:FD 36 00 01 3A 45 BF FE:70
B0C0:02 C2 D3 B0 FD 36 04 06:84
B0C8:3A 43 BF C6 10 FD 77 02:88
B0D0:C3 DF B0 FD 36 04 FA 3A:BD
B0D8:43 BF D6 04 FD 77 02 3A:8C
B0E0:56 BF A7 CA F1 B0 FD 36:5A
B0E8:03 FA FD 36 01 74 C3 F9:61
B0F0:B0 FD 36 03 06 FD 36 01:20
B0F8:88 06 02 78 FD 21 4B BF:30
B100:1E 03 CD 60 BB FD 7E 00:84
B108:FE 04 D2 23 B1 3A 44 BF:E5
B110:4F FD 7E 01 C6 06 81 FD:15
B118:77 01 FE E0 DA 23 B1 FD:01
B120:36 00 07 05 78 FE FF C2:79
B128:FB B0 06 09 78 FD 21 05:55
B130:BF 1E 03 CD 60 BB 3A 44:46
B138:BF 4F FD 7E 01 81 FD 77:7F
B140:01 05 78 FE FF C2 2C B1:1A
B148:06 09 78 FD 21 23 BF 1E:A5
B150:03 CD 60 BB 3A 44 BF 4F:77
B158:FD 7E 01 81 FD 77 01 FE:70
B160:FC DA 74 B1 C5 CD 9F BC:E8
B168:C1 E6 03 FE 03 C2 71 B1:8F
B170:AF FD 77 00 05 78 FE FF:9D
B178:C2 4A B1 CD 64 B9 CD C5:39
B180:BA CD 21 B8 CD 6C B4 01:4E
B188:04 FF CD FC B1 FE AA CA:EF
B190:9A B1 3E 03 32 41 BF C3:81
B198:30 B2 01 04 10 CD FC B1:71
B1A0:FE AA CA AD B1 3E 02 32:42
B1A8:41 BF C3 30 B2 01 10 04:BA
B1B0:C5 CD FC B1 E6 C0 FE 80:63
B1B8:CA C4 B1 C1 3E 04 32 41:B5
B1C0:BF C3 30 B2 C1 04 78 FE:9F
B1C8:0C CA CF B1 C3 B0 B1 01:7B
B1D0:FE 04 C5 CD FC B1 E6 C0:E7
B1D8:FE 80 CA E6 B1 C1 3E 05:E3
B1E0:32 41 BF C3 30 B2 C1 04:9C
B1E8:78 FE 0C CA F1 B1 C3 D2:83
B1F0:B1 3A 00 BF E6 01 32 41:04
B1F8:BF C3 30 B2 FD 21 03 BF:44
B200:3A 42 BF 80 67 3A 43 BF:5E
B208:81 6F CD D0 BB 56 23 5E:1F
B210:78 A7 CA 1F B2 7B 17 5F:AB
B218:7A 17 57 05 C3 10 B2 7A:EC
B220:C9 FD 21 03 BF 3A 47 BF:E9
B228:80 67 3A 48 BF C3 08 B2:A5
B230:3A 46 BF A7 CA 92 B2 3A:2E
B238:48 BF FE 38 DA 47 B2 FE:0E
B240:88 D2 47 B2 C3 8F B2 01:58
B248:00 00 CD 21 B2 FE AA C2:0A
B250:60 B2 01 00 03 CD 21 B2:B6
B258:FE AA C2 60 B2 C3 69 B2:5A
B260:3A 4A BF EE FF 3C 32 4A:E8
B268:BF 01 02 FF CD 21 B2 E6:47
B270:C0 FE C0 CA 86 B2 01 02:83
B278:04 CD 21 B2 E6 C0 FE C0:08
B280:CA 86 B2 C3 8F B2 3A 49:89
B288:BF EE FF 3C 32 49 BF CD:EF
B290:53 B4 2A 60 BF ED 4B 62:EA
B298:BF A7 ED 42 DA A2 B2 C3:86
B2A0:A6 B2 ED 43 60 BF 2A 62:33
B2A8:BF 11 1A 09 CD 21 BC 2A:C7
B2B0:60 BF 11 1A 0C CD 21 BC:00
B2B8:3A 43 BF FE 10 DA C8 B2:9E
B2C0:FE A8 D2 D0 B2 C3 D5 B2:44
B2C8:3E 10 32 43 BF C3 D5 B2:CC
B2D0:3E A8 32 43 BF CD 6D B3:07
B2D8:3A 44 BF 47 E6 80 C2 F9:A5
B2E0:B2 26 00 68 29 29 29 E5:A0
B2E8:D1 2A 64 BF A7 19 D8 19:CF
B2F0:D8 19 D8 22 64 BF C3 24:F5
B2F8:B3 26 00 78 EE FF 3C 6F:E9
B300:29 29 29 E5 D1 2A 64 BF:7E
B308:A7 ED 52 DA 1E B3 ED 52:D0
B310:DA 1E B3 ED 52 DA 1E B3:95
B318:22 64 BF C3 24 B3 3E 01:1E
B320:32 5D BF C9 3A 00 BF E6:F6
B328:01 CA 4D B3 3A 65 BF 0F:38
B330:0F E6 3F 47 3E C7 90 FE:0E
B338:C5 D2 4D B3 67 2E E0 11:1D
B340:48 B3 CD E8 BB C3 4D B3:2E
B348:01 03 44 54 44 CD 9F BC:08
B350:3A 28 FA FE 40 DA 4D B3:74
B358:3A 03 BF FE C0 CA 65 B3:9C
B360:3E 01 C3 67 B3 3E 02 CD:29
B368:ED 13 C3 90 AC 3A 41 BF:39
B370:FD 21 87 B3 1E 22 CD 60:C5
B378:BB FD E5 D1 3A 42 BF 67:10
B380:3A 43 BF 6F C3 6A BB 02:95
B388:10 02 80 82 80 82 80 32:C8
B390:82 0F FE 03 C0 03 C0 03:18
B398:C0 03 C2 03 C2 03 FE 0F:5A
B3A0:00 0C 00 B0 00 B0 00 80:EC
B3A8:00 02 10 02 80 02 82 02:1A
B3B0:82 82 8C BF F0 03 C0 03:05
B3B8:C0 03 C0 83 C0 83 C0 BF:C8
B3C0:C0 00 F0 00 30 00 0E 00:EE
B3C8:0E 00 02 02 10 02 80 02:A6
B3D0:80 02 80 02 80 0F F0 33:B6
B3D8:CC C3 C3 C3 C3 23 C8 23:E6
B3E0:C8 03 C0 0C 30 30 0C C0:C3
B348:03 38 2C 08 20 02 10 0C:AD
B3F0:30 0E B0 32 8C C2 83 32:23
B3F8:8C 0F F0 03 C0 03 C0 03:14
B400:C0 03 C0 0C 30 0C 30 30:2B
B408:0C 30 0C 80 02 80 02 02:4E
B410:10 00 30 28 30 28 C0 28:AB
B418:C2 2B 02 0F CE 0F FC 3F:16
B420:F0 33 C0 C3 C0 C0 C0 00:E6
B428:30 00 30 00 0C 00 0E 00:7A
B430:02 02 10 0C 00 0C 28 03:57
B438:28 B3 28 80 EB B3 F0 3F:1D
B440:F0 0F FC 03 CC 03 C3 03:93
B448:03 0C 00 0C 00 30 00 80:FB
B450:00 80 00 3A 46 BF A7 C8:2E
B458:3A 47 BF 57 3A 48 BF 6F 57
B460:11 66 B4 C3 6A B6 01 04:18
B468:14 14 14 14 06 02 78 FD:CD
B470:21 4B BF 1E 03 CD 60 BB:34
B478:FD 7E 00 FE 07 CA 8C B4:6A
B480:FD 66 01 FD 6E 02 FE 04:D3
B488:D2 A7 B4 FD 21 C4 B4 1E:E1
B490:44 CD 60 BB 3A 00 BF E6:0B
B498:01 CA A1 B4 11 22 00 FD:50
B4A0:19 FD E5 D1 C3 B7 B4 3D:37
B4A8:3D 3D 3D FD 21 D4 B5 1E:7C
B4B0:22 CD 60 BB FD E5 D1 C5:82
B4B8:CD E8 BB C1 05 78 FE FF:AB
B4C0:C2 6E B4 C9 02 10 AF EA:58
B4C8:AF FA B3 F2 BF FE FC FE:05
B4D0:CF FF F3 FF FF FF FF FF:BC
B4D8:FF CF FF F3 BF 3F BF FE:7B
B4E0:8F CE AF FA AB FA 02 10:BD
B4E8:AB FA AB FA AF CE 8F FE:54
B4F0:BF 3F BF F3 FF CF FF FF:7C
B4F8:FF FF F3 FF CF FE FC FE:B7
B500:BF F2 B3 FA AF EA AF EA:90
B508:02 10 9A A0 92 A0 A0 A0:BE
B510:A8 20 AA 05 6A 80 6A 44:0F
B518:4A 10 82 44 A0 18 A8 48:C8
B520:AA 19 A8 29 68 AA 50 AA:B0
B528:42 AA 02 10 AA 82 6A 82:16
B530:4A 82 82 82 A0 16 A8 00:2E
B538:AA 44 6A 10 40 48 40 18:48
B540:AA 09 AA 29 A6 2A 98 AA:9A
B548:90 AA 92 AA 02 10 88 88:98
```

```
B550:08 88 80 00 80 00 80 00:90
B558:80 00 80 00 80 00 85 14:19
B560:85 14 87 1C 87 1C 80 00:5F
B568:A0:02 A0 02 AB 0A 02 10:08
B570:A2 22 A2 22 80 00 80 00:88
B578:80 00 80 00 80 00 80 00:00
B580:85 14 85 14 8D 34 8D 34:B4
B588:80 00 A0 02 A0 02 AB 0A:76
B590:02 10 00 2A 15 2A 04 0A:89
B598:04 0A 04 02 04 42 04 40:9E
B5A0:04 00 00 54 00 40 80 40:5B
B5A8:80 54 A0 40 A0 40 A8 54:90
B5B0:A8 00 02 10 00 2A 3F 2A:4D
B5B8:0C 0A 0C 0A 0C 02 0C C2:08
B5C0:0C C0 0C 00 00 FC 00 C0:94
B5C8:80 C0 80 FC A0 C0 A0 C0:7C
B5D0:A8 FC A8 00 02 10 AA 6A:72
B5D8:AA 6A 99 AA AA 66 A6 6A:77
B5E0:9A 9A A9 A6 5A 6A A9 A5:95
B5E8:9A 6A A6 A6 A9 9A 99 AA:D6
B5F0:AA 66 A9 AA A9 AA 02 10:C8
B5F8:AA AA AA AA AA AA AA AA:50
B600:AA 6A A6 AA AA 9A A9 AA:FB
B608:AA 6A A6 AA AA 9A A9 AA:FB
B6A0:AA AA AA AA AA AA AA AA:50
B618:02 10 AA AA AA AA AA AA:0E
B620:AA AA AA AA AA AA AA AA:50
B628:AA AA AA AA AA AA AA AA:50
B630:AA AA AA AA AA AA AA AA:50
B638:AA AA 3A 42 BF D6 04 67:D0
B640:3A 43 BF D6 04 6F 01 03:89
B648:18 C3 B0 BC 3A 47 BF D6:5D
B650:0A 67 3A 48 BF D6 04 6F:FB
B658:01 02 18 C3 B0 BC 06 02:52
B660:78 FD 21 4B BF 1E 03 CD:8E
B668:60 BB FD 7E 01 D6 0A 67:DE
B670:FD 6E 02 C5 01 02 1A CD:1C
B678:B0 BC C1 05 78 FE FF C2:69
B680:60 B6 C9 21 10 10 01 14:35
B688:08 CD B0 BC 21 10 CC 01:3F
B690:14 04 C3 B0 BC DD 21 00:45
B698:BF ED 5B 01 BF D5 FD 7E:17
B6A0:01 C6 08 67 FD 6E 02 01:A4
B6A8:02 08 DD 36 01 C2 DD 36:F3
B6B0:02 DA DD 36 03 C0 DD E5:74
B6B8:C5 E5 CD B0 BC E1 C1 DD:62
B6C0:E1 DD 36 01 E2 DD 36 02:EC
B6C8:FA DD 36 03 E0 DD E5 CD:7F
B6D0:B0 BC DD E1 D1 ED 53 01:3C
B6D8:BF 1D 1D DD 73 03 C9 00:15
B6E0:3E E2 32 01 BF 3E FA 32:7C
B6E8:02 BF 3E E0 32 03 BF CD:A0
B6F0:21 B8 CD 4A B8 CD 5C B8:89
B6F8:CD 53 B8 CD 23 B7 CD 70:BC
B700:B8 3E C2 32 01 BF 3E DA:C2
B708:32 02 BF 3E C0 32 03 BF:E5
B710:CD 21 B8 CD 4A B8 CD 5C:9E
B718:B8 CD 53 B8 CD 23 B7 CD:04
B720:70 B8 C9 21 19 00 DD 21:29
B728:C9 B7 CD 07 BD 21 10 01:4C
B730:DD 21 D1 B7 CD 07 BD 21:38
B738:19 02 DD 21 D9 B7 CD 07:7D
B740:BD 21 19 03 DD 21 E1 B7:90
B748:CD 07 BD 21 19 04 DD 21:CD
B750:E9 B7 CD 07 BD 21 19 05:70
B758:DD 21 F1 B7 CD 07 BD 21:58
B760:19 06 DD 21 F9 B7 CD 07:A1
B768:BD 21 19 07 DD 21 C9 B7:7C
B770:CD 07 BD 21 19 08 DD 21:D1
B778:01 B8 CD 07 BD 21 19 09:8D
B780:DD 21 09 B8 CD 07 BD 21:17
B788:19 0A DD 21 11 B8 CD 07:BE
B790:BD 21 19 0B DD 21 19 B8:D1
B798:CD 07 BD 21 19 0C DD 21:D5
B7A0:09 C8 CD 07 BD 21 19 0D:99
B7A8:DD 21 E1 B7 CD 07 BD 0E:35
B7B0:0E 3E 35 61 2E 19 CD E4:DA
B7B8:BC 3E 36 61 2E 1F CD E4:8F
B7C0:BC 0C 79 FE 17 C8 C3 B1:92
B7C8:B7 30 2E 2E 2E 2E 2E 31:FE
B7D0:FF 2F 1D 18 20 0E 1B 2F:DB
B7D8:FF 2F 19 0A 17 12 0C 2F:B5
B7E0:FF 32 2E 2E 2E 2E 2E 33:4A
B7E8:FF 30 1D 25 0E 2E 2E 31:0C
B7F0:FF 2F 1C 18 0F 1D 24 2F:E1
B7F8:FF 32 2E 01 09 08 02 33:A6
B800:FF 2F 1C 0C 18 1B 0E 2F:C6
B808:FF 2F 24 24 24 24 24 2F:11
B810:FF 2F 2E 2E 2E 2E 2E 2F:43
B818:FF 2F 11 12 10 11 27 2F:C8
B820:FF 3A 02 BF 47 3A 01 BF:3B
B828:67 2E 00 E5 FD E1 11 20:89
B830:00 FD 36 00 FF FD 36 01:66
B838:FF FD 36 17 FF FD 36 18:93
B840:FF FD 19 19 7C B8 C8 C3:ED
B848:31 B8 21 10 10 01 14 C0:FF
B850:C3 B0 BC 21 D0 80 01 04:A5
B858:48 C3 B0 BC 3E 55 32 D2:0E
B860:BC 21 C8 18 01 06 C0 CD:51
B868:B0 BC 3E AA 32 D2 BC C9:DD
B870:21 C8 B0 11 88 B8 CD EB:A2
B878:BB 21 E0 85 11 D5 B8 CD:AC
B880:E8 BB 21 E8 B0 11 1A B9:40
B888:C3 E8 BB 03 18 6A A9 6A:FE
B890:6A AA 2A 6A A8 AA 6A AA:0E
B898:2A 6A A9 AA 6A AA 6A 6A:CF
B8A0:A9 21 68 AA A2 A6 AA 69:FB
B8A8:68 A1 A6 6A AA 68 69 A4:38
B8B0:A6 68 4A 65 6A 84 A8 6A:BD
B8B8:69 22 69 86 89 64 99 26:26
B8C0:6A 66 49 66 91 A1 68 6A:83
B8C8:1A 61 98 68 69 A4 86 62:70
B8D0:69 59 6A AA AA 01 43 96:5A
B8D8:96 44 44 44 44 44 44 44:72
B8E0:44 44 44 44 44 44 44 44:20
B8E8:44 44 44 44 44 44 44 44:20
B8F0:44 44 44 44 44 44 44 44:20
B8F8:44 44 44 44 44 44 44 44:20
B900:44 44 44 44 44 44 44 44:20
B908:44 44 44 44 44 44 44 44:20
B910:44 44 44 44 44 44 44 44:20
B918:44 AA 03 18 AA AA A5 AA:AC
B920:A8 A5 6A AA A5 AA A8 A5:FD
B928:6A 4A A5 AA 98 A5 2A 69:D3
B930:65 96 99 A5 66 2A 65 8A:B8
B938:98 A5 A6 6A 65 99 89 95:79
B940:28 6A 25 96 9A 63 62 19:C7
B948:A5 99 96 65 66 99 A5 29:16
B950:26 25 9A 99 A5 68 62 25:12
B958:86 A9 A5 68 65 A5 25 92:FD
B960:65 AA AA A5 06 09 78 FD:E2
B968:21 05 BF 1E 03 CD 60 BB:EE
B970:FD 7E 00 FD E5 FD 21 99:14
B978:69 1E 02 CD 60 BB FD 56:14
B980:00 FD 5E 01 FD E1 FD 66:9D
B988:01 FD 6E 02 C5 CD E8 BB:A3
B990:C1 05 78 FE FF C2 66 B9:1C
B998:C9 B9 9F B9 E9 BA 4B 03:CB
B9A0:18 AA AA AA AA AA AA AA:BE
B9A8:AA AA AA AA AA AA AA AA:50
B9B0:AA AA AA AA AA AA AA AA:50
B9B8:AA FF FF FF FF FF FF FF:A3
B9C0:FF FF FF FF FF FF FF FF:F8
B9C8:FF FF FF FF FF FF FF FF:F8
B9D0:FF AA AA AA AA AA AA AA:A5
B9D8:AA AA AA AA AA AA AA AA:50
B9E0:AA AA AA AA AA AA AA AA:50
B9E8:AA 04 18 AA AA AA AA AA:18
B9F0:AA AA AA AA AA AA AA AA:50
B9F8:AA AA AA AA AA AA AA AA:50
BA00:AA AA AA AA AA AA AA AA:50
BA08:AA AA AA FF FF FF FF FF:F5
BA10:FF FF FF FF FF FF FF FF:F8
BA18:FF FF FF FF FF FF FF FF:F8
BA20:FF FF FF FF FF FF FF FF:F8
BA28:FF FF FF AA AA AA AA AA:4F
BA30:AA AA AA AA AA AA AA AA:50
BA38:AA AA AA AA AA AA AA AA:50
BA40:AA AA AA AA AA AA AA AA:50
```

```
BA48:AA AA AA 05 18 AA AA AA:19
BA50:AA AA AA AA AA AA AA AA:50
BA58:AA AA AA AA AA AA AA AA:50
BA60:AA AA AA AA AA AA AA AA:50
BA68:AA AA AA AA AA AA AA AA:50
BA70:AA AA AA AA AA FF FF FF:4F
BA78:FF FF FF FF FF FF FF FF:F8
BA80:FF FF FF FF FF FF FF FF:F8
BA88:FF FF FF FF FF FF FF FF:F8
BA90:FF FF FF FF FF FF FF FF:F8
BA98:FF FF FF FF FF AA AA AA:F9
BAA0:AA AA AA AA AA AA AA AA:50
BAA8:AA AA AA AA AA AA AA AA:50
BAB0:AA AA AA AA AA AA AA AA:50
BAB8:AA AA AA AA AA AA AA AA:50
BAC0:AA AA AA AA AA 06 09 78:D9
BAC8:FD 21 23 BF 1E 03 CD 60:4E
BAD0:BB FD 7E 00 FE 02 CA F4:F4
BAD8:BA FD E5 FD 21 FC BA 1E:8E
BAE0:32 CD 60 BB FD E5 D1 FD:CA
BAE8:E1 FD 66 01 FD 6E 02 C5:77
BAF0:CD E8 BB C1 05 78 FE FF:A8
BAF8:C2 C7 BA C9 03 10 AA AA:73
BB00:AA AA AA AA AA AA AA AA:50
BB08:AA AA AA AA AA AA AA AA:50
BB10:AA AA AA AA AA AA A8 00:A4
BB18:2A A0 44 0A 81 55 42 04:34
BB20:44 00 01 55 00 00 44 40:1E
BB28:05 55 00 80 44 02 03 10:33
BB30:AA AA AA AA AA AA AA AA:50
BB38:AA AA AA AA AA AA AA AA:50
BB40:AA AA AA AA AA AA AA AA:50
BB48:AA 2A AA AA 2A 8A A2 2A:A8
BB50:8A A2 2A 4A A0 2A 42 A4:50
BB58:0A 12 85 51 14 15 54 55:C4
BB60:A7 C8 16 00 FD 19 3D C3:9B
BB68:60 BB DD 21 01 BF FD 21:F7
BB70:03 BF 1A 4F 13 1A 47 13:B2
BB78:C5 CD D0 BB 78 C1 F5 7C:C7
BB80:DD BE 00 DA C6 BB A7 DD:7A
BB88:BE 01 D2 C6 BB CA C6 BB:5D
BB90:F1 E5 C5 47 C5 1A 13 D5:A9
BB98:EB 67 2E 00 05 FA AA BB:E4
BBA0:A7 7C 1F 67 7D 1F 6F C3:77
BBA8:9C BB EB 7E AA 77 23 7E:82
BBB0:AB 77 D1 C1 0D C2 94 BB:D2
BBB8:78 C1 E1 05 C8 D5 11 20:ED
BBC0:00 19 D1 C3 7E BB 79 3D:9C
BBC8:13 C2 C7 BB F1 C3 BB BB:81
BBD0:7D E6 07 47 A7 0E 03 7C:E5
BBD8:1F 67 7D 1F 6F A7 0D C2:07
BBE0:D7 BB 7C FD B6 00 67 C9:F1
BBE8:DD 21 01 BF FD 21 03 BF:9E
```

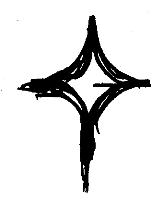
```
BBF0:CD D4 BB 1A 4F 13 1A 47:39
BBF8:13 E5 C5 7C DD BE 00 DA:AE
BC00:0E BC A7 DD BE 01 D2 0E:ED
BC08:BC CA 0E BC 1A 77 13 23:17
BC10:0D C2 FB BB C1 E1 05 C8:F4
BC18:D5 11 20 00 19 D1 C3 F9:AC
BC20:BB D5 CD 5F BC E1 78 0F:E0
BC28:0F 0F 0F E6 0F C5 E5 CD:59
BC30:E4 BC E1 C1 78 E6 0F 2C:DB
BC38:C5 E5 CD E4 BC E1 C1 79:32
BC40:0F 0F 0F 0F E6 0F 2C C5:22
BC48:E5 CD E4 BC E1 C1 79 E6:53
BC50:0F 2C C5 E5 CD E4 BC E1:33
BC58:C1 AF 2C CD E4 BC C9 11:E3
BC60:83 BC 01 00 00 7C A7 F8:5B
BC68:29 29 29 DA 76 BC 7C B5:B8
BC70:C8 13 13 C3 6A BC 1A 81:72
BC78:27 4F 13 1A 88 27 47 13:AC
BC80:C3 6A BC 92 81 96 40 48:1A
BC88:20 24 10 12 05 56 02 28:EB
BC90:01 64 00 32 00 16 00 08:85
BC98:00 04 00 02 00 01 00 3A:41
BCA0:04 BF 47 3A 2B FA 80 47:2D
BCA8:87 87 80 3C 32 04 BF C9:88
BCB0:DD 21 01 BF FD 21 03 BF:9E
BCB8:C5 CD D4 BB C1 0C E5 C5:98
BCC0:7C DD BE 00 DA D9 BC A7:2D
BCC8:DD BE 01 D2 D9 BC CA D9:A6
BCD0:BC 3E AA 77 23 0D C2 C0:CD
BCD8:BC C1 E1 05 C8 11 20 00:5C
BCE0:19 C3 BE BC F5 3A 01 BF:45
BCE8:84 67 F1 FD 21 3C BD 1E:11
BCF0:08 CD 60 BB 06 08 11 20:2F
BCF8:00 FD 7E 00 EE FF 77 FD:DC
BD00:23 19 05 C8 C3 F9 BC DD:5E
BD08:7E 00 FE FF C8 E5 CD E4:D9
BD10:BC E1 2C 7D FE 20 C8 DD:09
BD18:23 C3 07 BD 7E F5 23 7E:BE
BD20:D3 A0 23 7E D3 A1 F1 3D:B6
BD28:C8 C3 1D BD 21 33 BD CD:43
BD30:1C BD C9 04 07 FF 08 40:F4
BD38:09 40 0A 40 AA 56 66 66:5F
BD40:66 66 66 56 AA 9A 9A 9A:00
BD48:9A 9A 9A 9A AA 56 66 A6:74
BD50:9A 6A 6A 56 AA 56 66 A6:D0
BD58:96 A6 66 56 AA 6A 66 66:D8
BD60:66 56 A6 A6 AA 56 6A 6A:DC
BD68:9A A6 66 56 AA 56 66 6A:CC
BD70:56 66 66 55 AA 56 A6 A6:C4
BD78:A6 A6 A6 A6 AA 56 66 66:64
BD80:56 66 66 56 AA 56 66 66:44
BD88:56 A6 66 56 AA 9A 66 66:C8
BD90:56 66 66 66 AA 5A 66 66:58
```

```
BDA0:6A 6A 66 9A AA 5A 66 66:A4
BDA8:66 66 66 5A AA 56 6A 6A:60
BDB0:56 6A 6A 56 AA 56 6A 6A:54
BDB8:56 6A 6A 6A AA 9A 66 6A:A8
BDC0:6A 66 66 96 AA 66 66 66:A8
BDC8:56 66 66 66 AA 56 9A 9A:BC
BDD0:9A 9A 9A 56 AA 56 9A 9A:58
BDD8:9A 9A 9A 6A AA 66 66 5A:08
BDE0:5A 66 66 66 AA 6A 6A 6A:74
BDE8:6A 6A 6A 56 AA 66 56 56:50
BDF0:56 56 66 66 AA 66 56 56:34
BDF8:56 56 56 66 AA 9A 66 66:78
BE00:66 66 66 9A AA 5A 66 66:9C
BE08:5A 6A 6A 6A AA 9A 66 66:A8
BE10:66 66 66 96 AA 5A 66 66:98
BE18:5A 66 66 66 AA 9A 66 6A:A0
BE20:9A A6 66 9A AA 56 9A 9A:74
BE28:9A 9A 9A 9A AA 66 66 66:44
BE30:66 66 66 9A AA 66 66 66:A8
BE38:66 66 9A 9A AA 66 66 56:CC
BE40:56 56 56 66 AA 66 66 66:44
BE48:9A 66 66 66 AA 66 66 66:A8
BE50:66 9A 9A 9A AA 56 A6 A6:80
BE58:9A 6A 6A 56 AA AA AA AA:6C
BE60:AA AA AA AA AA AA AA 9A:40
BE68:56 66 66 66 AA AA AA AA:30
BE70:56 AA AA AA AA AA AA AA:FC
BE78:AA AA 6A 6A AA AA AA AA:D0
BE80:AA 9A 9A 6A AA AA 9A 9A:D0
BE88:AA 9A 9A AA AA AA 9A 9A:10
BE90:56 9A 9A AA AA AA AA 56:88
BE98:AA 56 AA AA AA A6 9A 6A:A8
BEA0:6A 6A 9A A6 AA 6A 9A A6:68
BEA8:A6 A6 9A 6A AA AA AA AA:F8
BEB0:00 AA AA AA 8A 8A 8A 8A:26
BEB8:8A 8A 8A 8A AA AA AA AA:D0
BEC0:80 8A 8A 8A AA AA AA AA:C6
BEC8:0A 8A 8A 8A 8A 8A 8A 8A:D0
BED0:80 AA AA AA 8A 8A 8A 8A:A6
BED8:0A AA AA AA 9A 9A 56 56:E8
BEE0:9A 56 66 66 95 95 95 95:10
BEE8:95 95 95 95 5A 5A 5A 5A:BC
BEF0:5A 5A 5A 5A 00 00 00 00:68
BEF8:00 00 00 00 00 00 00 00:00
```

IE 80

立體飛行模擬

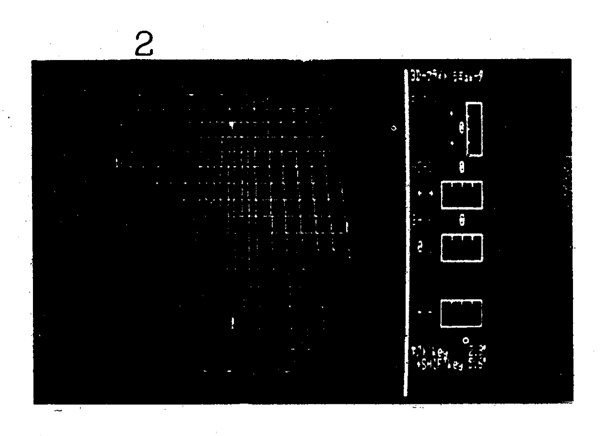
NEC 8801



也許您聽過飛行員講述他們飛行的樂趣。滔滔不絕地講述他們如何駕着銀鷹在萬里無雲在空中飛翔，極目望去，一覽衆山小的感覺。當你駕着飛機着陸時，將看到機場上五顏六色的指示燈和線條，在你的面前將呈現出一幅五光十色的圖畫。可惜我們大家都沒有駕過飛機的經驗。如今我們可以利用電腦產生這些五光十色的圖象，它富有高度的立體感和眞實性，足可使你感慨地說〝百聞不如一見〞了。現在就讓我們一起飛行吧。這些圖象是採用彩色3維坐標表示的飛行模擬裝置，卽稱爲Colour 3D flig-ht simulator。

1

2



在這三維（3D）空間中，若有二個機場，就可以從一個機場定期地飛到另一個機場，這種定期飛行，就有許多相遇的機會，也可以比賽它們飛行的速度等等。這些飛機將眞實地給你操作控制，你可任意駕駛這些飛機在美麗的彩色世界中飛翔。

飛行方法

（1）在MON狀態中，用 r RETURN 輸入（LOAD）機器語言。

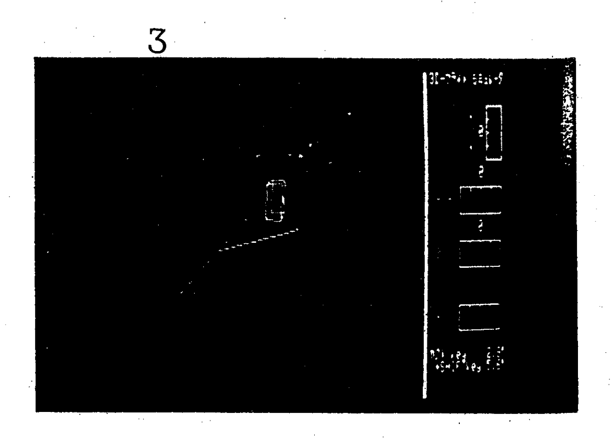
（2）由 cn+1 b 退到BASIC程序中，並用BASIC語言工作（LOAD）（load〝cas O:〞 RETURN ）

（3）由run RETURN 慢慢地開始飛行了。

（4）當你按下 SPACE 鍵，在屏幕上就出現如照片1所示的遊戲標題。

（5）圖畫上的內容一轉，就出現了飛機場和周圍的風景。如照片2所示。

3



由照片中，你可看出在遠方的地平線附近所出現的白點，這就是飛機場，以前的遊戲用黑白圖象表示，那麽這些白色點就埋在地平線裏，不容易看出來，這也算是彩色圖象的優點了。以前在飛行中看不到什麽東西，只有在離機場不遠的地方才看到着陸點。也就是說，對於太遠的點子是看不清楚的。

飛機停在機場上，由於定期飛行時間表在作怪，它會從這裏再出發。首先提高速度在機場上滑行。鍵盤上的十個鍵中的⊞ 鍵是增加速度的（SPEED UP）鍵，要增加速度就按⊞鍵使之上升。按着你就可以自如地駕駛飛機了。有關各種操作鍵的功能，請參考表1。

爲了提高速度，可使用機器語言，若按上 STOP 鍵，就會退到BASIC狀態。

地形數據

有關地形如圖1所示，其南北方向爲飛機場，可作爲定期班次飛行。它的東、西、南、北方向是山峰。

用BASIC語言編寫出標題和初始設定值。這部分可以自己根據需要進行改寫的。它的主程序部分是用機器語言寫成的。它的存儲單元的分配如表2所示。由於存取圖象的程序必須寫在&ch BFFF以前，故應該把它們分開，其中&ch CO9-9 &c CC6F間爲未使用區域。

飛機場等的三維生標數據由&ch CEOO輸入，因此，若改變這部分數據，就可以改變圖形的形狀，也可以改變其顏色。

變更數據時的具體輸入形式，可參見表3的內容進行判斷變更。它如一筆畫成的那樣，由起點——中間點——中間點……起始點——中間點……終點的順序進行，它可輸入的線條數沒有固定，但一般可達到180條左右。（其中要多留有一些處理的時間）

程序的補充說明

主程序的方框原理流程如圖3所示。有關三坐標圖形部分模塊爲&ch D430～&chDE8F，而&ch D430～&ch D43D爲控制表。由於它和參考文獻1中的3D通用模塊是相同的，故可參考這部分內容。（其中，這個模塊所對應的彩色是專業應用的。故通用模塊本身不能更改使用）。

由於圖象的清除程序速度很高，故R、G、B的每一個圖象均能方便地清除，若高速寫入狀態（Sc-reen，1）的程序作爲CALL時，則在一瞬間就可馬上消除。

線條LINE程序最初用640×200點作出任意兩點的連線，其X方向的分辨率爲640點，要用2byte來作爲數據處理。在高速的模擬中，判斷方向，1byte數據的上限可用256×256點的線條LINE。由於圖象的分辨率爲640×200點，故在X橫方向用2點同一數據來表示，在Y方向用上下各28點來表示，在圖象上的表示區域爲512×200點。

這個遊戲的模擬速度是很快的，你不會感到有任何的遲頓。由於PC－8801的存儲單元可以變換，故能承擔過重的負載，而且可使用高速處理和高速顯示的程序。盡管如此，其實還沒有達到本電腦應用的極限。假如你不相信，那就可以試試看。

備 註

一

一

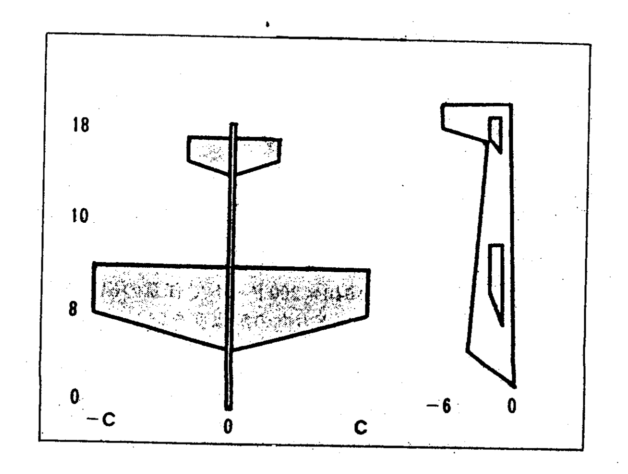
按上左邊記號的按鍵，其角度變化2.8°，若同時按上SH-IFT鍵其角度變化5.6°（度）

表1 使用鍵（KEY）

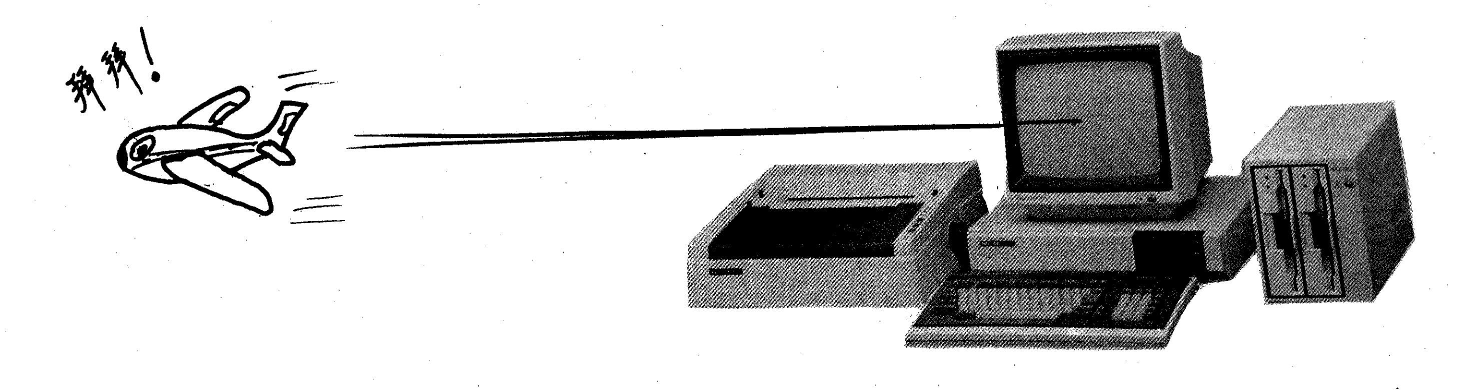
		使用KEY
SPEED	UP	+（十個鍵）
SPEED	DOWN	－（十個鍵）
PITCH	上	↑
PITCH	下	↓
BANK	順時針方向	，（十個鍵）
BANK	反時針方向	0（十個鍵）
HEAD	右	→
HEAD	左	←

*同時按上二個以上的鍵工作也正常

飛機場和定期班次飛機的圖形如圖2所示。



```
10 '**********************
20 '* 3D FLIGHT SIMULATER *
30 '**********************
40 CLEAR ,&HBCFF
50 DEF USR0=&HBDA0:DEF USR1=&HBF50
60 DIM G%(123)
70 SCREEN 0,0:WIDTH 80,25:CONSOLE 0,25,0,1
80 COLOR 4,1:CLS 3
90 COLOR 4
100 PRINT" ■■■■■      ■■"
110 PRINT"■     ■      ■"
120 PRINT"■      ■■  ■ ■■ ■ ■"
130 PRINT"■     ■ ■ ■ ■ ■■ ■"
140 PRINT"■     ■ ■ ■ ■ ■"
150 PRINT"■    ■ ■ ■ ■ ■ ■ "
160 PRINT" ■■■■■  ■■  ■ ■■ ■"
170 PRINT
180 COLOR 6
190 PRINT"■■■■■■■■ ■■     ■     ■■■■■■   ■      ■■"
200 PRINT"■     ■     ■  ■  ■      ■      ■   ■"
210 PRINT"■     ■ ■■  ■■ ■■■ ■■■ ■     ■■ ■■ ■ ■ ■ ■ ■■■ ■■■
 ■■  ■ ■"
220 PRINT"■■■■■   ■ ■ ■ ■ ■ ■ ■   ■■■■■■  ■ ■ ■ ■ ■ ■ ■  ■ ■
■ ■ ■■ ■"
230 PRINT"■     ■ ■ ■ ■ ■ ■ ■      ■ ■ ■ ■ ■ ■ ■ ■ ■■■ ■
■■■ ■  ■     ■ ■ ■■■ ■ ■ ■  ■   ■ ■ ■ ■ ■ ■■ ■ ■ ■ ■
■   ■"
240 PRINT"■     ■ ■■■  ■ ■ ■ ■■   ■■■■■■ ■■■ ■ ■ ■  ■ ■ ■ ■■ ■ ■■
 ■■■ ■       ■"
250 PRINT"       ■■"
260 COLOR 7:PRINT
270 PRINT
280 SADD=&HCC70
290 FOR I=0 TO 123
300  A1$=RIGHT$("0"+HEX$(PEEK(SADD)),2):A2$=RIGHT$("0"+HEX$(PEEK(SADD+1)),2)
310  G%(I)=VAL("&H"+A1$+A2$):SADD=SADD+2
320 NEXT
330 FOR X=600 TO 110 STEP -4:PUT (X,58),G%,PSET:NEXT
340 LOCATE 40,20:PRINT"                    ":LOCATE 40,22:PRINT"
 "
346 A$=INKEY$:IF A$="" THEN 346
350 CLS:A%=USR1(0)
355 SCREEN ,1
360 FOR I=0 TO 9:READ A:POKE &HD430+I,A:NEXT
370 DATA &h80,&hff,0,0,&he0,&h0b,0,0,0,0
390 COLOR 6:LOCATE 65,2:PRINT "PITCH":LOCATE 65,7:PRINT"HEAD":LOCATE 65,11:PRINT
 "BANK"
400 COLOR 4:LOCATE 65,16:PRINT "SPEED"
410 COLOR 5:LOCATE 75,2:PRINT "┌──┐":FOR Y=3 TO 5:LOCATE 75,Y:PRINT "├  │":NEXT:
LOCATE 75,Y:PRINT "└──┘"
420 X=70:Y=8:GOSUB 490:X=70:Y=12:GOSUB 490:X=70:Y=17:GOSUB 490
430 COLOR 3:LOCATE 72,3:PRINT " ":LOCATE 72,5:PRINT " ":LOCATE 66,9:PRINT "   ":
LOCATE 65,13:PRINT "       ":LOCATE 66,18:PRINT "+ -"
440 POKE &HF578,30:POKE &HF668,31:POKE &HF842,29:POKE &HF844,28
445 LOCATE 66,13:PRINT"0 ,"
450 COLOR 7:LOCATE 74,4:PRINT "0":LOCATE 74,7:PRINT "0":LOCATE 74,11:PRINT "0"
455 LOCATE 65,21:PRINT "    key   :2.8°":LOCATE 66,22:PRINT "+SHIFTkey:5.6°"
460 FOR Y=0 TO 24:LOCATE 64,Y:PRINT "▌";:NEXT
470 A%=USR0(0)
480 SCREEN 0,0:COLOR ,0:CLS 3:END
490 LOCATE X,Y:PRINT"┌┬┬┬┬┬┐":LOCATE X,Y+1:PRINT"│       │":LOCATE X,Y+2:PRINT
"└───────┘":RETURN
```



```
ADD  +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F SUM
BDA0 ED 73 4D BF 31 9F BD 21 00 CE 22 3A D4 21 90 DE :A7
BDB0 22 3C D4 DB 09 FE FE CA EE BE D6 08 47 0E 02 CB :8D
BDC0 70 20 02 0C 0C CB 48 20 07 3A 36 D4 81 32 36 D4 :E5
BDD0 CB 50 20 07 3A 38 D4 81 32 38 D4 DB 0A 47 CB 48 :86
BDE0 20 07 3A 36 D4 91 32 36 D4 CB 50 20 07 3A 38 D4 :C0
BDF0 91 32 38 D4 CB 68 20 0B 3A 39 D4 FE 00 28 04 3D :DB
BE00 32 39 D4 DB 00 CB 47 20 07 3A 37 D4 91 32 37 D4 :66
BE10 D6 01 47 CB 68 20 07 3A 37 D4 81 32 37 D4 CB 58 :A3
BE20 20 0B 3A 39 D4 FE 3F 28 04 3C 32 39 D4 3A 39 D4 :9D
BE30 08 3A 36 D4 CD FB D5 E5 08 26 00 6F 08 CD 6C D4 :80
BE40 ED 5B 32 D4 19 22 32 D4 3A 3B D4 CD FB D5 E5 08 :5F
BE50 26 00 6F 08 CD 6C D4 ED 5B 30 D4 AF EB ED 52 22 :F1
BE60 30 D4 E1 D1 CD 6C D4 08 16 00 5F CD 6C D4 ED 5B :95
BE70 34 D4 AF EB ED 52 22 34 D4 CD 1F D6 CD 50 BF CD :76
BE80 00 C0 3A FF D2 A7 F2 8E BE FE F3 28 66 37 08 21 :8F
BE90 FD D2 06 14 56 23 5E 08 38 0B 08 7B 0E 0A 91 5F :96
```

```
SUM  A4 6C B1 15 F0 93 D7 C7 F4 B0 36 7F E4 3E F2 7C :E0
ADD  +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F SUM
BEA0 30 06 15 18 06 08 7B C6 0A 5F 30 01 14 2B 72 23 :27
BEB0 73 23 23 7E D6 20 77 30 05 2B 7E 3D 77 23 23 23 :9F
BEC0 23 23 10 D0 08 3A 36 D4 21 8D D0 CD 00 BF 3A 38 :EE
BED0 D4 21 C6 D6 CD 1B BF 3A 37 D4 21 C6 E0 CD 1B BF :EB
BEE0 3A 39 D4 C6 04 21 46 ED CD 21 BF C3 B3 BD ED 7B :AD
BEF0 4D BF C9 21 70 CD 11 FA D2 01 8C 00 ED B0 18 C5 :17
BF00 CB 2F CB 2F CB 2F C6 10 47 AF 11 50 00 ED 52 10 :6A
BF10 FC F3 D3 5D 3E FF 77 D3 5F FB C9 CB 2F CB 2F C6 :63
BF20 23 5F E6 07 E5 21 45 BF 85 6F 7E E1 CB 3B CB 3B :D8
BF30 CB 3B 16 00 19 11 50 00 06 08 F3 D3 5E 77 19 10 :68
BF40 FC D3 5F FB C9 80 40 20 10 08 04 02 01 FF 00 FF :EF
BF50 AF C5 D5 E5 F3 D3 5C CD 6B BF D3 5D CD 6B BF D3 :41
BF60 5E CB 6B BF D3 5F FB E1 D1 C1 C9 21 00 C0 11 01 :B1
BF70 C0 01 FF 3F 77 ED B0 C9 20 20 20 28 48 4C 29 2C :4D
BF80 C5 D5 E5 AF 47 7B D6 1C 38 66 FE C8 30 62 4F 7A :A1
BF90 16 00 CB 27 30 01 14 5F E6 07 21 F4 BF 85 6F 7E :DF
```

```
SUM  7A 61 93 6A AB E6 41 9F C1 43 14 C7 68 0E 0B 55 :3E
ADD  +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F SUM
BFA0 CB 21 CB 10 CB 21 CB 10 CB 21 CB 10 CB 21 CB 10 :1C
BFB0 C5 CB 21 CB 10 CB 21 CB 10 E1 09 C6 3A CB 1B CB :F3
BFC0 3A CB 1B CB 3A CB 1B 19 11 00 C0 19 5F 3A 99 C0 :00
BFD0 57 F3 CB 52 26 05 D3 5C 7B B6 77 CB 4A 28 05 D3 :80
BFE0 5D 7B B6 77 CB 42 28 05 D3 5E 7B B6 77 D3 5F FB :45
BFF0 E1 D1 C1 C9 C0 00 30 00 0C 00 03 75 D4 AE BE 00 :F0
C000 21 90 DE E5 E1 7E CB 7F C0 32 99 C0 23 56 23 5E :62
C010 23 7E 23 4E 23 E5 69 CD 1C C0 18 E8 67 92 26 54 :A1
C020 30 03 EB ED 44 47 0E 00 7D 93 26 61 30 05 0D ED :6C
C030 44 18 01 0C B8 30 1E 6F 7C 92 60 47 7C CB 3F 08 :21
C040 CD 80 8F 14 08 94 85 30 07 08 7B 81 5F 38 18 01 :FC
C050 84 06 10 EC C9 67 CB 3F 08 7D 93 30 02 ED 44 68 :A5
C060 47 CD 80 BF 7B 81 5F 08 94 85 30 03 14 18 01 84 :83
C070 08 10 EE C9 7B 95 26 10 30 0A ED 44 47 CD 80 BF :D5
C080 1C 10 FA C9 5D 47 18 F5 5D CD 80 BF C9 7A 94 38 :18
C090 01 54 CD 80 BF 14 10 FA C9 00 00 00 00 00 00 00 :48
```

```
SUM  D4 E8 3A 33 AB 44 91 86 14 0E 6D F1 B4 DB A9 F4 :DD
ADD  +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F SUM
CBA0 FF 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF 00 :FB
CBB0 FF 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF 00 :FB
CBC0 FF 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF 00 :FB
CBD0 FF 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF 00 :FB
CBE0 FF 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF 00 :FB
CBF0 FF 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF 00 :FB
CC00 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF :FB
CC10 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF :FB
CC20 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF :FB
CC30 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF :FB
CC40 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF :FB
CC50 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF :FB
CC60 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF 00 FF :FB
CC70 00 24 00 10 FF FF FF F0 00 FF 0F 00 08 00 00 00 :37
CC80 00 00 FF 08 E1 FF FF FF 00 00 00 1E 00 08 00 00 :0B
CC90 08 00 FF FF FF C3 00 FF 3C 00 08 00 00 00 00 00 :0B
```

```
SUM  68 18 05 30 0E 8C CA C9 D8 D0 30 62 C5 07 32 38 :28
ADD  +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F SUM
CDA0 C0 F0 00 80 00 08 08 C5 25 00 8C 00 0A 0B C8 25 :BE
CDB0 00 8C 00 0A 0B CC 25 00 80 00 0B 0B CC 25 00 74 :8D
CDC0 00 0A 0B CC 25 00 74 00 0A 0B C8 25 00 80 00 0B :07
CDD0 0B C5 F0 00 80 00 08 0B D4 25 00 84 00 06 0B D5 :BE
CDE0 25 00 84 00 0B 06 D7 25 00 7C 00 0B 0B D7 25 00 :49
CDF0 7C 00 0B 0B D5 25 00 80 00 0B 0B D4 FF FF FF FF :F2
CE00 F0 00 90 00 10 F4 F6 23 00 70 00 10 F4 F6 23 00 :2E
CE10 70 00 10 F4 0B 23 00 90 00 10 F4 08 23 00 50 00 :EE
CE20 10 F4 4A 23 00 E4 00 10 F4 9E 23 00 CE 00 10 F4 :EC
CE30 B4 23 00 90 00 10 F4 76 23 00 50 00 10 F4 F8 F0 :80
CE40 00 A0 00 10 F4 C8 25 00 B8 00 10 F4 C8 25 00 B8 :F2
CE50 00 10 F4 F0 25 00 A0 00 10 F4 F0 25 00 A0 00 10 :82
CE60 F4 C8 25 00 A0 FF E0 F4 C8 25 00 B8 FF E0 F4 C8 :94
CE70 25 00 B8 FF E0 F4 F0 25 00 A0 FF E0 F4 F0 25 00 :4D
CE80 A0 FF E0 F4 C8 F0 00 B8 00 10 F4 C8 25 00 B8 FF :8B
CE90 ED F4 C8 F0 00 B8 00 10 F4 F0 25 00 B8 FF E0 F4 :E8
```

```
SUM  02 1D F6 10 D9 8A F8 E7 36 F8 11 17 02 01 FA F9 :E5
ADD  +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F SUM
CCA0 FF 0B 87 FF FF FF 00 00 00 7B 00 08 00 00 08 00 :13
CCB0 FF FF F3 0F 00 FF F0 00 08 0C 00 00 00 00 FF 08 :0A
CCC0 0F FE FF E3 01 00 1C F0 00 08 00 00 08 00 FC FF :07
CCD0 E3 0F 00 FF F0 05 08 1C 00 00 00 00 00 08 00 00 :10
CCE0 FF 01 FF FF FE FF 00 08 00 00 FB 01 00 00 01 00 :FD
CCF0 FF FF FF FF 08 FE 00 00 01 00 FF F8 0F FC FF E3 :E7
CD00 03 00 1C F0 00 08 00 00 08 02 FE FF E3 0F 00 FF :0D
CD10 F0 01 08 1C 00 00 00 00 FF 06 0F FF FF F3 00 00 :1C
CD20 0C F0 00 08 00 00 08 00 FF FF FF 87 00 FF 87 00 :07
CD30 08 00 00 00 00 00 FF 08 C3 FF FF FF 00 00 00 3C :0B
CD40 00 08 00 00 08 00 FF FF FF E1 00 FF 1E 00 08 00 :13
CD50 00 00 00 00 FF 08 F0 FF FF FF 00 00 00 0F 00 08 :0B
CD60 00 00 08 00 00 00 00 00 00 00 00 00 00 00 00 00 :08
CD70 F0 00 80 00 0C 08 C0 23 00 80 00 08 0B C3 23 00 :E3
CD80 80 00 0A 0B D5 23 00 80 00 06 0B D6 23 00 80 00 :97
CD90 86 0B D8 23 00 80 00 0C 0B D8 23 00 80 00 0C 0B :35
```

```
SUM  29 CD ED EB 05 75 07 8F 1E 8E 29 24 6D 11 63 0F :9B
ADD  +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F SUM
CEA0 F0 F0 00 A0 00 10 F4 F0 25 00 A0 FF E0 F4 F0 F0 :EC
CEB0 00 80 FF E0 F4 E8 26 00 B0 FF D0 F4 E8 26 00 B6 :C8
CEC0 FF D0 F4 E8 26 00 B6 FF D6 F4 E8 26 00 B0 FF D6 :E3
CED0 F4 E8 F0 00 94 00 10 F5 10 22 00 94 00 08 F5 10 :38
CEE0 F0 00 6C 00 10 F5 10 22 00 6C 00 08 F5 10 F0 00 :FC
CEF0 94 00 10 F4 D0 22 00 94 00 08 F4 D0 F0 00 6C 00 :46
CF00 10 F4 D0 22 00 6C 00 08 F4 D0 F0 00 94 00 10 F4 :B6
CF10 90 22 00 94 00 08 F4 90 F0 00 6C 00 10 F4 90 22 :E4
CF20 00 6C 00 08 F4 90 F0 00 80 00 10 F4 08 27 00 80 :1B
CF30 00 10 F4 FB F0 00 80 00 10 F4 50 27 00 D8 00 10 :CF
CF40 F4 A8 F0 F0 00 00 10 F0 00 24 10 00 00 10 F0 00 :B0
CF50 F0 F0 00 00 10 F2 00 24 10 00 00 10 F2 00 F0 F0 :F8
CF60 00 00 10 F4 00 24 10 00 00 10 F4 00 F0 F0 00 00 :1C
CF70 10 F6 00 24 10 00 00 10 F6 00 F0 F0 00 00 10 F8 :28
CF80 00 24 10 00 00 10 F8 00 F0 F0 00 00 10 FA 00 24 :4A
CF90 10 00 00 10 FA 00 F0 F0 00 00 10 FC 00 24 10 00 :3A
```

```
SUM  0B 9C 33 2A 8C 39 5C 46 25 71 0C 9C 4B F3 E0 3E :05
ADD  +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F SUM
CFA0 00 10 FC 00 F0 F0 00 00 10 FE 00 24 10 00 00 10 :3E
CFB0 FE 00 F0 F0 00 00 10 00 00 24 10 00 00 10 00 00 :32
CFC0 F0 F0 00 00 10 02 00 24 10 00 00 10 02 00 F0 F0 :1B
CFD0 00 00 10 04 00 24 10 00 00 10 04 00 F0 F0 00 00 :3C
CFE0 10 06 00 24 10 00 00 10 06 00 F0 F0 00 00 10 08 :5B
CFF0 00 24 10 00 00 10 08 00 F0 F0 00 00 00 0A 00 24 :6A
D000 10 00 00 10 0A 00 F0 F0 00 00 10 0C 00 24 10 00 :5A
D010 00 10 0C 00 F0 F0 00 00 10 0E 00 24 10 00 00 10 :5E
D020 0E 00 F0 F0 00 00 10 10 00 24 10 00 00 10 10 00 :62
D030 F0 F0 00 00 10 F0 00 24 F0 00 00 10 10 00 F0 F2 :F6
D040 00 00 10 F0 00 24 F2 00 00 10 10 00 F0 F4 00 00 :1A
D050 10 F0 00 24 F4 00 00 10 10 00 F0 F6 00 00 10 F0 :1E
D060 00 24 F6 00 00 10 10 00 F0 F8 00 00 10 F0 00 24 :46
D070 F8 00 00 10 10 00 F0 FA 00 00 10 F0 00 24 FA 00 :20
D080 00 10 10 00 F0 FC 00 00 10 F0 00 24 FC 00 00 10 :3C
D090 10 00 F0 FE 00 00 10 F0 00 24 FE 00 00 10 10 00 :40
```

```
SUM  24 4E 0E 3A 0E 36 2A 52 26 70 32 6E 2E 56 2A 52 :B0
ADD  +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F SUM
D0A0 F0 00 00 00 10 F0 00 24 00 00 00 10 10 00 F0 02 :26
D0B0 00 00 10 F0 00 24 02 00 00 10 10 00 F0 04 00 00 :3A
D0C0 10 F0 00 24 04 00 00 10 10 00 F0 06 00 00 10 F0 :3E
D0D0 00 24 06 00 00 10 10 00 F0 08 00 00 10 F0 00 24 :66
D0E0 08 00 00 10 10 00 F0 0A 00 00 10 F0 00 24 0A 00 :50
D0F0 00 10 10 00 F0 0C 00 00 10 F0 00 24 0C 00 00 10 :5C
D100 10 00 F0 0E 00 00 10 F0 00 24 0E 00 00 10 10 00 :60
D110 F0 10 00 00 10 F0 00 24 10 00 00 10 10 00 F0 FD :41
D120 00 00 10 F0 00 24 FD 00 00 10 10 00 F0 FF 00 00 :30
D130 10 F0 00 24 FF 00 00 10 10 00 F0 01 00 00 10 F0 :34
D140 00 24 01 00 00 10 10 00 F0 03 00 00 10 F0 00 24 :5C
D150 03 00 00 10 10 00 F0 F2 00 00 10 10 00 21 F6 00 :3C
D160 FD 10 10 00 21 FA 00 00 10 10 00 21 FC 00 FE 10 :83
D170 10 00 21 FF 00 00 10 10 00 F0 F8 99 FF 10 10 00 :F0
D180 21 FA 00 FE 80 10 00 21 FA CA FF 30 10 00 F0 F0 :AD
D190 00 00 10 02 00 21 F0 00 FE 80 04 80 21 F0 00 00 :36
```

```
SUM  49 52 68 55 D4 7F 0F 85 28 89 29 B5 58 38 0E 37 :A3
ADD  +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F SUM
D1A0 10 06 00 21 F0 00 FE 00 09 00 21 F0 00 00 10 0C :5B
D1B0 00 F0 10 00 00 10 03 00 21 10 00 FE 30 01 00 21 :94
D1C0 10 00 00 10 FE 80 21 10 00 FE 00 FD 00 21 10 00 :FB
D1D0 00 10 FB 00 F0 0C 00 00 10 F0 00 21 0A 00 FE 80 :B0
D1E0 F0 00 21 07 00 00 10 F0 00 21 04 80 FD 60 F0 00 :0A
D1F0 21 00 00 00 10 F0 00 21 FE 80 FF 00 F0 00 21 FD :CD
D200 00 00 10 F0 00 F0 00 80 00 10 0B 08 26 00 80 00 :39
D210 10 0B F8 F0 00 90 00 [illegible] 0B 08 25 00 90 00 10 0B :86
D220 F8 25 00 70 00 10 0B F8 25 00 70 00 10 0B 08 25 :7D
D230 00 90 00 10 0B 08 F0 00 94 00 10 0A F0 22 00 94 :F7
D240 00 08 0A F0 F0 00 6C 00 10 [illegible] F0 22 00 6C 00 08 :FE
D250 0A F0 F0 00 94 00 10 0B 30 22 00 94 00 08 0B 30 :C2
D260 F0 00 6C 00 10 0B 30 22 00 6C 00 08 0B 30 F0 00 :6B
D270 94 00 10 0B 70 22 00 94 00 08 0B 70 F0 00 6C 00 :B4
D280 10 0B 70 22 00 6C 00 08 0B 70 F0 00 60 00 10 0B :07
D290 A0 21 00 60 00 10 0B D0 21 00 48 00 10 0B D0 21 :81
```

```
SUM  77 EA 1A 15 FD CD E4 42 68 C7 07 CC 4B 5E 0E D2 :08
ADD  +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F SUM
D2A0 00 48 00 10 0B A0 21 00 60 00 10 0B A0 21 00 60 :C0
D2B0 FF E0 0B A0 21 00 60 FF E0 0B D0 21 00 48 FF E0 :0D
D2C0 0B D0 21 00 48 FF E0 0B A0 21 00 60 FF E0 0B A0 :D9
D2D0 F0 00 48 00 10 0B A0 21 00 48 FF E0 0B A0 F0 00 :D6
D2E0 60 00 10 0B D0 21 00 60 FF E0 0B D0 F0 00 48 00 :BE
D2F0 10 0B D0 21 00 48 FF E0 0B D0 F0 00 80 00 0C 0B :95
D300 C0 23 00 80 00 08 0B C3 23 00 80 00 0A 0B D5 23 :E9
D310 00 80 00 06 0B D6 23 00 80 00 06 0B D8 23 00 80 :96
D320 00 0C 0B D8 23 00 80 00 0C 0B C0 F0 00 80 00 0B :E4
D330 0B C5 25 00 8C 00 0A 0B C8 25 00 8C 00 0A 0B CC :F0
D340 25 00 80 00 0B 0B CC 25 00 74 00 0A 0B CC 25 00 :26
D350 74 00 0A 0B C8 25 00 80 00 0B 0B C5 F0 00 80 00 :41
D360 0B 0B D4 25 00 84 00 0B 0B D5 25 00 84 00 0B 0B :3D
D370 D7 25 00 7C 00 0B 0B D7 25 00 7C 00 0B 0B D5 25 :16
D380 00 80 00 0B 0B D4 FF FF FF FF FF FF FF FF FF FF :60
D390 FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF :F0
```

```
SUM  AF 26 E1 F0 EB 83 8D BE 8F A6 CA 90 84 76 B1 93 :2C
ADD  +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F SUM
D3A0 FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF :F0
D3B0 FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF :F0
D3C0 FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF :F0
D3D0 FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF :F0
D3E0 FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF :F0
D3F0 FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF :F0
D400 FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF :F0
D410 FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF :F0
D420 FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF :F0
D430 00 00 00 00 00 00 00 00 00 00 00 D0 90 DE 00 00 :3E
D440 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 :00
D450 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 :00
D460 00 00 00 00 00 00 00 00 00 00 00 00 06 00 CD E3 :B6
D470 D5 EB CD E3 D5 C5 44 4D CD B2 D4 C1 78 A7 FC EA :84
D480 D5 C9 21 00 00 79 0F 30 01 19 CB 1C CB 1D 0F 30 :9F
D490 01 19 B7 CB 1C CB 1D 0F 30 01 19 B7 CB 1C CB 1D :7F
```

```
SUM  A2 C4 9C A5 EB 00 67 83 F5 93 AF 5B 9B B5 9A 11 :06
ADD  +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F SUM
D4A0 0F 30 01 19 B7 CB 1C CB 1D 0F 30 01 19 B7 CB 1C :D6
D4B0 CB 1D 0F 30 01 19 B7 CB 1C CB 1D 0F 30 01 19 B7 :D7
D4C0 CB 1C CB 1D 0F 30 01 19 78 A7 20 08 CB 15 CB 14 :2E
D4D0 6C 26 00 C9 CB 1C CB 1D 0F 30 01 19 B7 CB 1C CB :EC
D4E0 1D 0F 30 01 19 B7 CB 1C CB 1D 0F 30 01 19 B7 CB :D7
D4F0 1C CB 1D 0F 30 01 19 B7 CB 1C CB 1D 0F 30 01 19 :3C
D500 B7 CB 1C CB 1D 0F 30 01 19 B7 CB 1C CB 1D 0F D0 :44
D510 19 C9 06 00 CD E3 D5 EB CD E3 D5 C5 42 4B CD 24 :20
D520 D5 C3 7B D4 11 00 00 EB 29 EB A7 ED 42 30 02 09 :08
D530 1B 13 29 EB 29 EB A7 ED 42 30 02 09 1B 13 29 EB :A9
D540 29 EB A7 ED 42 30 02 09 1B 13 29 EB 29 EB A7 ED :0F
D550 42 30 02 09 1B 13 29 EB 29 EB A7 ED 42 30 02 09 :E4
D560 1B 13 29 EB 29 EB A7 ED 42 30 02 09 1B 13 29 EB :A9
D570 29 EB A7 ED 42 30 02 09 1B 13 29 EB 29 EB A7 ED :0F
D580 42 30 02 09 1B 13 29 EB 29 EB A7 ED 42 30 02 09 :E4
D590 1B 13 29 EB 29 EB A7 ED 42 30 02 09 1B 13 29 EB :A9
```

```
SUM  16 2F 92 8B 0B 21 D3 25 B3 FB 35 17 51 EB 2E 40 :27
ADD  +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F SUM
D5A0 29 EB A7 ED 42 30 02 09 1B 13 29 EB 29 EB A7 ED :0F
D5B0 42 30 02 09 1B 13 29 EB 29 EB A7 ED 42 30 02 09 :E4
D5C0 1B 13 29 EB 29 EB A7 ED 42 30 02 09 1B 13 29 EB :A9
D5D0 29 EB A7 ED 42 30 02 09 1B 13 29 EB C9 7C BA C0 :26
D5E0 7D BB C9 7C A7 F0 78 EE 80 47 97 95 6F 3E 00 9C :B6
D5F0 67 C9 7D 93 6F 7C 9A 67 FE 80 C9 6F 26 00 A7 F2 :A1
D600 03 D6 25 29 01 90 DD 09 56 23 5E C6 40 6F 26 00 :10
D610 A7 F2 15 D6 25 29 01 90 DD 09 46 23 6E 60 C9 2A :73
D620 3A D4 22 62 D4 2A 3C D4 22 64 D4 3A 38 D4 CD FB :08
D630 D5 22 44 D4 EB 22 3E D4 3A 36 D4 CD FB D5 22 46 :77
D640 D4 EB 22 40 D4 3A 37 D4 CD FB D5 22 48 D4 EB 22 :22
D650 42 D4 2A 44 D4 EB 2A 48 D4 CD 6C D4 E5 2A 3E D4 :B7
D660 EB 2A 40 D4 CD 6C D4 EB 2A 42 D4 CD 6C D4 D1 19 :58
D670 22 50 D4 2A 44 D4 EB 2A 42 D4 CD 6C D4 E5 2A 3E :0D
D680 D4 EB 2A 40 D4 CD 6C D4 EB 2A 48 D4 CD 6C D4 D1 :19
D690 CD F2 D5 22 52 D4 2A 3E D4 EB 2A 46 D4 CD 6C D4 :54
```

```
SUM  10 71 8E F6 A2 D5 F4 C3 7A C1 FB 09 D3 50 75 8C :C6
ADD  +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F SUM
D6A0 22 54 D4 2A 46 D4 EB 2A 42 D4 CD 6C D4 22 56 D4 :12
D6B0 2A 46 D4 EB 2A 48 D4 CD 6C D4 22 58 D4 2A 40 D4 :0E
D6C0 CD EA D5 22 5A D4 2A 3E D4 EB 2A 48 D4 CD 6C D4 :56
D6D0 E5 2A 44 D4 EB 2A 40 D4 CD 6C D4 EB 2A 42 D4 CD :55
D6E0 6C D4 D1 CD F2 D5 22 5C D4 2A 3E D4 EB 2A 42 D4 :5E
D6F0 CD 6C D4 E5 2A 44 D4 EB 2A 40 D4 CD 6C D4 EB 2A :7F
D700 48 D4 CD 6C D4 D1 19 22 5E D4 2A 44 D4 EB 2A 46 :04
D710 D4 CD 6C D4 22 60 D4 2A 62 D4 7E FE F0 20 10 CD :00
D720 B2 D7 ED 43 3E D4 ED 53 40 D4 22 42 D4 18 EB 57 :AE
D730 E6 F8 FE 20 28 06 2A 64 D4 36 FF C9 7A 08 CD B2 :8B
D740 D7 ED 43 44 D4 ED 43 4A D4 ED 53 46 D4 ED 53 4C :53
D750 D4 22 48 D4 22 4E D4 CD 5A D8 30 41 2A 64 D4 08 :30
D760 77 23 E5 ED 5B 3E D4 2A 42 D4 CD 4C D8 E1 77 23 :85
D770 E5 ED 5B 40 D4 2A 42 D4 CD 4C D8 E1 77 23 E5 ED :BF
D780 5B 44 D4 2A 48 D4 CD 4C D8 E1 77 23 E5 ED 5B 46 :98
D790 D4 2A 48 D4 CD 4C D8 E1 77 23 22 64 D4 2A 4A D4 :28
```

```
SUM  21 EB 71 A3 67 01 F5 95 AD 04 89 20 15 F0 1A E1 :6C
ADD  +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F SUM
D7A0 22 3E D4 2A 4C D4 22 40 D4 2A 4E D4 22 42 D4 C3 :FB
D7B0 17 D7 23 56 23 5E E5 2A 30 D4 19 22 4A D4 E1 23 :58
D7C0 56 23 5E E5 2A 32 D4 19 22 4C D4 E1 23 56 23 5E :22
D7D0 23 22 62 D4 2A 34 D4 19 22 4E D4 ED 5B 4A D4 2A :9A
D7E0 50 D4 CD 6C D4 E5 ED 5B 4C D4 2A 56 D4 CD 6C D4 :DF
D7F0 D1 19 E5 ED 5B 4E D4 2A 5C D4 CD 6C D4 D1 19 E5 :6F
D800 ED 5B 4A D4 2A 52 D4 CD 6C D4 E5 ED 5B 4C D4 2A :3A
D810 58 D4 CD 6C D4 D1 19 E5 ED 5B 4E D4 2A 5E D4 CD :9B
D820 6C D4 D1 19 E5 ED 5B 4A D4 2A 54 D4 CD 6C D4 E5 :B9
D830 ED 5B 4C D4 2A 5A D4 CD 6C D4 D1 19 E5 ED 5B 4E :32
D840 D4 2A 60 D4 CD 6C D4 D1 19 D1 C1 C9 CD 12 D5 11 :49
D850 7F 00 D5 CD 6C D4 D1 19 7D C9 06 00 CD 42 D9 38 :B7
D860 02 06 01 CD 4E D9 38 04 78 F6 02 47 CD 59 D9 38 :27
D870 04 78 F6 04 47 CD 64 D9 38 04 78 F6 08 47 0E 00 :CE
D880 ED 5B 44 D4 2A 48 D4 19 7C FE 80 38 02 0E 01 2A :2C
D890 48 D4 CD F2 D5 38 04 79 F6 02 4F ED 5B 46 D4 2A :38
```

```
SUM  FF 7C DA F7 CC 9B A5 43 41 01 6E 5F 95 9F 72 26 :76
ADD  +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F SUM
D8A0 48 D4 CD F2 D5 38 04 79 F6 04 4F 2A 48 D4 19 7C :89
D8B0 FE 80 38 04 79 F6 08 4F 78 A1 C0 78 B1 37 C8 79 :FA
D8C0 B8 30 02 48 47 78 07 07 07 07 B1 01 21 00 21 DF :E0
D8D0 B8 B7 ED B1 C0 21 00 D9 09 09 5E 23 56 EB E9 1E :C2
D8E0 2D 4B 5A 4A 1A 69 49 29 16 25 24 14 12 78 68 58 :CE
D8F0 48 28 18 0F 0E 0D 0B 0A 09 08 07 06 05 04 02 01 :F1
D900 4B DB 62 DB 1D DB 6C DB 82 DB EE DB 34 DB 77 DB :29
D910 8D DB D8 DB 09 DC F3 DB 0E DC A4 DB BC DB D0 DB :79
D920 27 DB 39 DB 0E DC C8 DB 98 DB B0 DB 67 DB 3E DB :FC
D930 55 DB 22 DB EC DA 50 DB 10 DB C8 DA 0E DC 0E DC :7F
D940 0E DC ED 5B 3E D4 2A 42 D4 19 7C FE 80 C9 ED 5B :A8
D950 3E D4 2A 42 D4 CD F2 D5 C9 ED 5B 40 D4 2A 42 D4 :4B
D960 CD F2 D5 C9 ED 5B 40 D4 2A 42 D4 19 7C FE 80 C9 :D5
D970 ED 5B 3E D4 2A 44 D4 22 3E D4 ED 53 44 D4 ED 5B :70
D980 40 D4 2A 46 D4 22 40 D4 ED 53 46 D4 ED 5B 42 D4 :46
D990 2A 48 D4 22 42 D4 ED 53 48 D4 C9 DC 70 D9 2A 44 :36
```

```
SUM  0F 33 23 56 DC E0 3B 7B 0F 92 FA A5 5D D8 F0 23 :B5
ADD  +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F SUM
D9A0 D4 22 66 D4 2A 46 D4 22 68 D4 2A 48 D4 22 6A D4 :78
D9B0 C9 CD DD D5 C0 E1 E1 E1 E5 11 5A D7 CD DD D5 28 :79
D9C0 01 E1 A7 C9 2A 3E D4 CD EA D5 22 3E D4 2A 44 D4 :90
D9D0 CD EA D5 22 44 D4 C9 2A 40 D4 CD EA D5 22 40 D4 :8F
D9E0 2A 46 D4 CD EA D5 22 46 D4 C9 CD F2 D5 3F CB 1C :8F
D9F0 CB 1D 19 C9 CD 4E D9 CD 9B D9 EB 2A 44 D4 CD B1 :AA
DA00 D9 ED 5B 68 D4 2A 40 D4 CD EA D9 E5 ED 5B 6A D4 :96
DA10 2A 42 D4 CD EA D9 E5 ED 5B 66 D4 2A 3E D4 CD EA :2A
DA20 D9 D1 CD DD D5 20 0C 22 3E D4 E1 22 40 D4 ED 53 :E0
DA30 42 D4 C9 E5 CD F2 D5 E1 38 0D 22 66 D4 E1 22 68 :45
DA40 D4 ED 53 6A D4 18 BA 22 3E D4 E1 22 40 D4 ED 53 :AF
DA50 42 D4 18 AD CD C4 D9 CD F4 D9 CD C4 D9 C9 CD 59 :38
DA60 D9 CD 9B D9 EB 2A 46 D4 CD B1 D9 ED 5B 66 D4 2A :4C
DA70 3E D4 CD EA D9 E5 ED 5B 6A D4 2A 42 D4 CD EA D9 :DD
DA80 E5 ED 5B 68 D4 2A 40 D4 CD EA D9 D1 CD DD D5 20 :A7
DA90 0C 22 40 D4 E1 22 3E D4 ED 53 42 D4 C9 E5 CD F2 :1A
```

SUM	9C	62	DF	37	89	A8	97	97	A7	D0	A7	B4	80	D4	BB	AB	:FF
ADD	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	SUM
DAA0	D5	E1	38	0D	22	68	D4	E1	22	66	D4	ED	53	6A	D4	18	:2C
DAB0	BA	22	40	D4	E1	22	3E	D4	ED	53	42	D4	18	AD	CD	D7	:C4
DAC0	D9	CD	5E	DA	CD	D7	D9	C9	CD	5E	DA	2A	42	D4	2B	7C	:10
DAD0	FE	80	D0	CD	4E	D9	D0	CD	42	D9	D4	54	DA	CD	64	D9	:06
DAE0	D0	CD	BE	DA	CD	4E	D9	D4	F4	D9	37	C9	CD	F4	D9	2A	:8E
DAF0	42	D4	2B	7C	FE	80	D0	CD	64	D9	D0	CD	59	D9	D4	5E	:16
DB00	DA	CD	42	D9	D0	CD	54	DA	CD	64	D9	D4	BE	DA	37	C9	:03
DB10	CD	F4	D9	CD	59	D9	D0	CD	64	D9	D4	BE	DA	CD	5E	DA	:E4
DB20	37	C9	CD	54	DA	18	EC	CD	54	DA	CD	64	D9	D0	CD	59	:FA
DB30	D9	D4	5E	DA	CD	BE	DA	37	C9	CD	F4	D9	18	EC	CD	5E	:13
DB40	DA	CD	42	D9	D0	CD	4E	D9	D4	F4	D9	CD	54	DA	37	C9	:22
DB50	CD	BE	DA	18	EC	CD	BE	DA	CD	4E	D9	D0	CD	42	D9	D4	:4E
DB60	54	DA	CD	F4	D9	37	C9	CD	5E	DA	18	EC	CD	5E	DA	CD	:A3
DB70	42	D9	D4	54	DA	37	C9	CD	BE	DA	CD	42	D9	D4	54	DA	:6C
DB80	37	C9	CD	5E	DA	CD	4E	D9	D4	F4	D9	37	C9	CD	BE	DA	:FF
DB90	CD	4E	D9	D4	F4	D9	37	C9	CD	5E	DA	CD	42	D9	D0	CD	:1F

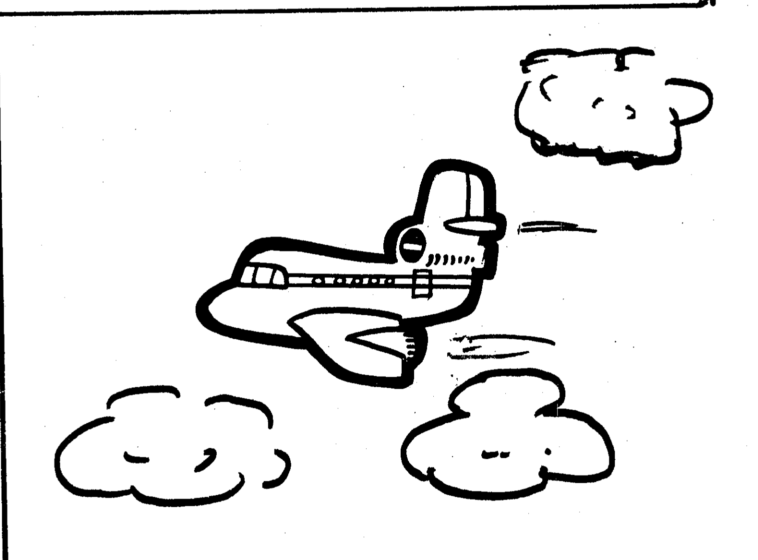
SUM	70	A4	38	1D	F6	32	71	86	22	CE	83	73	08	DC	D8	11	:3B
ADD	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	SUM
DBA0	54	DA	37	C9	CD	BE	DA	CD	42	D9	D0	CD	54	DA	37	C9	:46
DBB0	CD	5E	DA	CD	4E	D9	D0	CD	F4	D9	37	C9	CD	BE	DA	CD	:95
DBC0	4E	D9	D0	CD	F4	D9	37	C9	CD	F4	D9	CD	54	DA	37	C9	:26
DBD0	CD	5E	DA	CD	BE	DA	37	C9	CD	BE	DA	3A	43	D4	A7	FB	:8F
DBE0	CD	4E	D9	D4	F4	D9	CD	42	D9	D4	54	DA	37	C9	CD	5E	:AA
DBF0	DA	18	EB	CD	F4	D9	3A	43	D4	A7	F8	CD	59	D9	D4	5E	:95
DC00	DA	CD	64	D9	D4	BE	DA	37	C9	CD	54	DA	18	E8	ED	5B	:53
DC10	48	D4	2A	42	D4	CD	F2	D5	DC	70	D9	2A	46	D4	2B	2B	:B1
DC20	7C	FE	80	D0	2A	44	D4	22	66	D4	2A	46	D4	22	68	D4	:0A
DC30	2A	48	D4	22	6A	D4	ED	5B	3E	D4	2A	66	D4	CD	EA	D9	:F4
DC40	E5	ED	5B	40	D4	2A	68	D4	CD	EA	D9	E5	ED	5B	42	D4	:7A
DC50	2A	6A	D4	CD	EA	D9	2B	7C	B5	20	0F	23	22	42	D4	E1	:8F
DC60	22	40	D4	E1	22	3E	D4	C3	5A	DB	7C	A7	23	F2	7D	DC	:D1
DC70	22	42	D4	E1	22	40	D4	E1	22	3E	D4	18	B5	22	6A	D4	:95
DC80	E1	22	68	D4	E1	22	66	D4	18	AC	00	00	00	00	00	00	:40
DC90	00	00	FE	6D	FC	DC	FB	4A	F9	BA	F8	2A	F6	9B	F5	0E	:F1

SUM	DF	87	9B	EE	D0	1E	48	4C	D5	4A	B7	E5	31	DF	EC	B9	:11
ADD	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	SUM
DCA0	F3	83	F1	FA	F0	73	EE	EE	ED	6B	EB	EC	EA	70	E8	F7	:08
DCB0	D7	82	E6	10	E4	A2	E3	39	E1	D4	E0	74	DF	18	DD	C2	:A0
DCC0	DC	71	DB	25	D9	E0	D8	A0	D7	66	D6	32	D5	05	D3	DE	:4E
DCD0	D2	BE	D1	A5	D0	94	CF	89	CE	86	CD	8B	CC	98	CB	AC	:49
DCE0	CA	C9	C9	ED	C9	1A	C8	50	C7	8E	C6	D5	C6	25	C5	7D	:61
DCF0	C4	DF	C4	49	C3	BD	C3	3A	C2	C1	C2	51	C1	EB	C1	8E	:BE
DD00	C1	3A	C0	F1	C0	B1	C0	7B	C0	4E	C0	2C	C0	13	C0	04	:E9
DD10	C0	00	C0	04	C0	13	C0	2C	C0	4E	C0	7B	C0	B1	C0	F1	:AE
DD20	C1	3A	C1	8E	C1	EB	C2	51	C2	C1	C3	3A	C3	BD	C4	49	:16
DD30	C4	DF	C5	7D	C6	25	C6	D5	C7	8E	C8	50	C9	1A	C9	ED	:71
DD40	CA	C9	CB	AC	CC	98	CD	8B	CE	86	CF	89	D0	94	D1	A5	:4C
DD50	D2	BE	D3	DE	D5	05	D6	32	D7	66	D8	A0	D9	E0	DB	25	:91
DD60	DC	71	DD	C2	DF	18	E0	74	E1	D4	E3	39	E4	A2	E6	10	:84
DD70	E7	82	E8	F7	EA	70	EB	EC	ED	6B	EE	EE	F0	73	F1	FA	:FB
DD80	F3	83	F5	0E	F6	9B	F8	2A	F9	BA	FB	4A	FC	DC	FE	6D	:67
DD90	00	00	01	92	03	23	04	B5	06	45	07	D5	09	64	0A	F1	:01

SUM	6E	2C	6F	ED	73	17	75	A3	77	8F	7B	E3	7F	99	81	AB	:40
ADD	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	SUM
DDA0	0C	7C	0E	05	0F	8C	11	11	12	94	14	13	15	8F	17	08	:E8
DDB0	18	7D	19	EF	1B	5D	1C	C6	1E	2B	1F	8B	20	E7	22	3D	:50
DDC0	23	8E	24	DA	26	1F	27	5F	28	99	29	CD	2A	FA	2C	21	:A2
DDD0	2D	41	2E	5A	2F	68	30	76	31	79	32	74	33	67	34	53	:A7
DDE0	35	36	36	12	36	E5	37	AF	38	71	39	2A	39	DA	3A	82	:8F
DDF0	3B	20	3B	B6	3C	42	3C	C5	3D	3E	3D	AE	3E	14	3E	71	:32
DE00	3E	C5	3F	0E	3F	4E	3F	84	3F	B1	3F	D3	3F	EC	3F	FB	:07
DE10	40	00	3F	FB	3F	EC	3F	D3	3F	B1	3F	84	3F	4E	3F	0E	:44
DE20	3E	C5	3E	71	3E	14	3D	AE	3D	3E	3C	C5	3C	42	3B	B6	:DA
DE30	3B	20	3A	82	39	DA	39	2A	38	71	37	AF	36	E5	36	12	:7F
DE40	35	36	34	53	33	67	32	74	31	79	30	76	2F	68	2E	5A	:A4
DE50	2D	41	2C	21	2A	FA	29	CD	28	99	27	5F	26	1F	24	DA	:5F
DE60	23	8E	22	3D	20	E7	1F	8B	1E	2B	1C	C6	1B	5D	19	EF	:6C
DE70	18	7D	17	08	15	8F	14	13	12	94	11	11	0F	8C	0E	05	:F5
DE80	0C	7C	0A	F1	09	64	07	D5	06	45	04	B5	03	23	01	92	:89

* D090 ~ E66F SET為 00

SUM	84	C6	83	96	81	FD	80	03	80	A7	7D	E3	7B	BC	7A	37	:D3
E670	F2	F6	C7	92	AA	32	C7	7F	7F	EF	D7	D7	EF	FF	FF	EF	:BD
E680	F0	B8	58	58	B8	F8	F8	B8	D7	D7	EF	EF	D7	D7	EF	7F	:60
E690	58	58	B8	B8	58	58	B8	F0	7F	FF	FD	DA	AA	AD	DF	EF	:F2
E6A0	F0	D8	A8	A8	D8	F8	F8	B8	6F	D7	D7	EF	EF	D7	D7	EF	:30
E6B0	B0	58	58	B8	B8	58	58	B8	EF	D7	D7	EF	EF	D7	D7	6F	:D0
E6C0	B8	58	38	B8	B8	58	58	B0	7F	DD	AA	AA	DD	FF	DD	AA	:4B
E6D0	F0	D8	A8	A8	D8	F8	D8	A8	AA	DD	FF	DD	AA	AA	DD	7F	:7B
E6E0	A8	D8	F8	D8	A8	A8	D8	F0	7F	FF	FD	C0	FD	E0	ED	E0	:4D
E6F0	F0	F8	F8	18	F8	38	B8	38	ED	E0	F8	F5	ED	DD	FF	7F	:1A
SUM	05	7B	2E	C8	C0	51	33	76	F0	14	37	82	87	BF	29	F3	:30





穿牆過壁

16 K

ZX

此程式完全利用畫面的24行顯示，主程式是測試看看遊戲中的球是否失手。

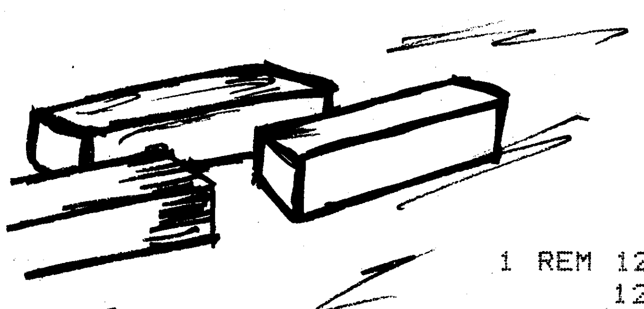
最下的一行是利用於球拍之用，行210是用於找出球的位置，從而得知是否失球，當變數爲1之時即表示球已擊中一磚塊，隨之即表示已經失球。每當變數A爲1之時，分數之數值亦相應加上一分。

球拍是以三個反白的空格，球亦只會被這3個空格反彈。球拍的位置可隨意改變，按8即向右，按5即向左。當全部的磚塊被擊去之後，新的一個畫面即重新出現。

本程式亦有使用機械碼的副程式，用家可嘗試放之REM之後，當BASIC程式輸入之前而BAS—IC本身亦有減慢的副程式。

機械碼的輸入程式如下：——

```
10 INPUT X
20 LET A$=""
30 IF A$="" THEN INPUT A$
40 IF A$="S" THEN STOP
50 POKE X,16*CODE A$+CODE A$(2)-476
60 LET X=X+1
70 LET A$=A$(3TO)
80 GOTO 30
```



以上程式是用於把機械碼放入REM之用，當輸入機械碼之時“／”是用以代表“NEWLINE”之用。

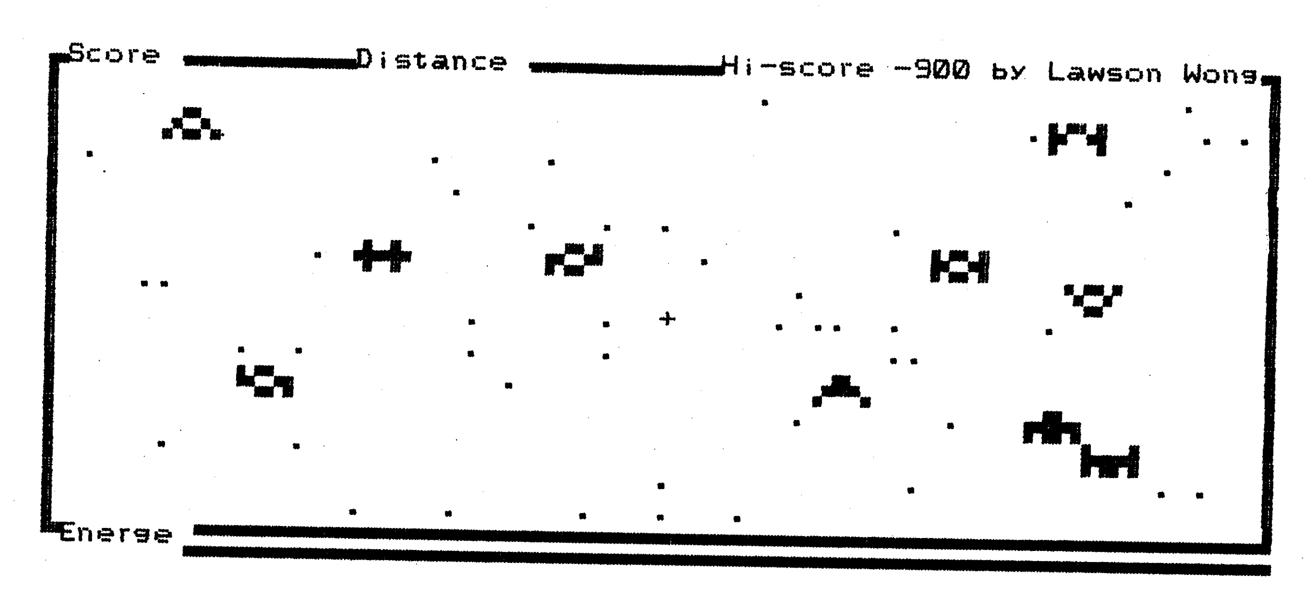
```
1 REM 1234567890123456789012345678901234567890
      1234567890123456789012312345678901234567
      8901234567890123200
```

```
16516/0101/2A8240/3600/3A8440/3D/2002/23/23/2B/
7E/FE80/200B/2A8240/4A8440/ED44/328440/228240/3A
8540/3D/2006/11DFFF/19/1804/112100/19/7E/FE80/20
0B/2A8240/3A8540/ED44/328540/010000/7E/FE08/2009
/03/3A8540/ED44/328540/228240/3634/C9/S
```

```
10 POKE 16418,0
20 LET TS=0
30 LET B=3
40 PRINT "32inverse spaces"
50 PRINT "inverse space,30 spaces, inverse space"
60 PRINT "inverse space,30 spaces, inverse space"
70 PRINT "inverse space,30 graphic H, inverse space"
80 PRINT "inverse space,30 graphic H, inverse space"
90 FOR I=1 TO 18
100 PRINT "inverse space,30  spaces, inverse space"
110 NEXT I
120 LET BP=200
130 PRINT "inverse space,30 graphic H, inverse space"
140 LET S=0
150 LET P=15
160 LET M= PEEK 16396 + 256 * PEEK 16397
170 LET X=M+BP
180 POKE 16514,X-256*INT (X/256)
190 POKE 16515,INT(X/256)
200 LET A=USR16518
210 IF A=1 AND (PEEK 16514 + 256* PEEK 16515 )>X THEN GOTO 300
220 IF A=1 THEN LET S=S+1
230 PRINT AT 21,P;"space,3 inverse spaces,space"
240 LET P=P+(INKEY$="8" AND P<26)-( INKEY$="5" AND P>1)
250 IF S=60 THEN GOTO  400
260 GOTO 200
300 FOR I=1 TO 75
310 NEXT I
320 LET TS=TS+S
330 LET B=B-1
340 LET BP=BP+INT(10*RND+1)
350 PRINT AT 21,P;"5 spaces"
360 IF B>0 THEN GOTO 130
370 PRINT AT 8,6;"YOU SCORES ";TS;"POINTS"
380 STOP
400 FOR I=1 TO 10
410 FOR J=1 TO 5
420 NEXT J
430 PRINT AT 10,10;"BONUS BALL"
440 FOR J=1 TO 5
450 NEXT J
460 PRINT AT 10,10;"BONUS BALL"
470 NEXT I
480 LET TS=TS+S
490 LET B=B+1
500 CLS
510 GOTO 40
```

SPACE ATTACK

GENIE I



今次地球面臨一次重大的危機，所有的太空怪物都一齊出動，計劃侵佔地球，統治人類，這豈不是人類的大悲劇呢！

熱血的好男兒們豈能坐以待斃，登上你們的太空飛船與入侵的太空怪物作一次大戰吧！

發動強烈的推動火箭，直冲雲霄，穿過星雲，深入敵方的陣地。因爲路途遙遠，能源大限，以有限對無限，以死一併，犧牲小我，而能保衛地球，可算是一件光榮的事情呢！

每分每秒緊握操操杆，把握每一個機會，珍惜每一分能源，盡你所能擊毀入侵的飛船，當你創上了新的殺敵記錄時，則名留千古，爲人歌頌，反之則變成無名英雄了！

祝大家好運！

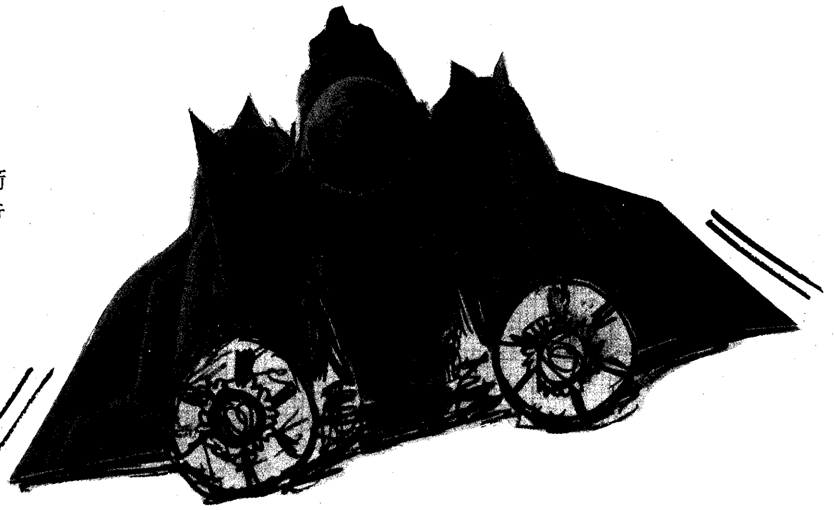
遊戲特點：提高速度和發射飛彈會加速消耗能源。敵人是不規則地移動運行，所以極不容易被擊中的。

遊戲中各按鍵的功能

（1）（↑）向上箭咀——向上
（2）（↓）向下箭咀——向下
（3）（→）向右箭咀——向右
（4）（←）向左箭咀——向左
（5）每一個以９０度角配合的箭咀，如：上箭咀及左箭咀同時按下卽會以４５度角飛行。
（6）（ＳＰＡＣＥ　ＢＡＲ）——發砲之用
（7）（ＣＬＥＡＲ）——加速向前

程式內容分析

行６０—１２０　：利用機械碼副程式去製造聲音效果。
行９２０—９７０：副程式（輸入玩者姓名與模擬進入飛船。
行１９０—４００：用ＲＡＮＤＯＭ方法去訂出怪物的距離及形狀。
行４２０—５１０：測試各按鍵。
行５３０—５７０：飛碟自已的軌跡。
行５９０—７３０：測試飛碟是否已離開攻擊範圍。
行７８０—９１０：是飛碟飛離射程及遊戲完畢時運行的副程式。



```
10 '******************************
20 '*  S p a c e   A t t a c k   *
30 '*    By Mr.Wong leung fung   *
40 '* 07/11/82 Edited programme *
50 '******************************
60 CLEAR1000:DIMR(50):ST=127
70 DATA 205,127,10,22,30,62,5,211,255,69,
        16,254,60,211,255,69,16,254,21,32,
        240,201
80 MUSIC$="Mirco special software"
90 A=VARPTR(MUSIC$):A1=PEEK(A+2):A2=PEEK(A+1)
100 FORZ=0 TO 21:
          READY:POKEZ+A1*256+A2,Y:
    NEXT
110 IFPEEK(16396)<>201,DEFUSR=A1*256+A2:CMD"T":GOTO130
120 POKE16526,A2:POKE16527,A1
130 N$=" By Lawson wong":S$="0900"
150 HL=0:CLS:GOSUB920
160 FORI=0 TO 8:
          PRINT@I*64,CHR$(30):
          PRINT@(15-I)*64,CHR$(30):
    NEXT
170 Z=0:FORI=64 TO 0 STEP-1:
                 MUSIC=USR(I):Z=Z+1:
                 SET(I,1):SET(I,45):SET(127-I,45):SET(127-I,1):
    NEXT
180 FORI=23 TO 2 STEP-1:
                 MUSIC=USR(I):Z=Z-1:
                 SET(0,I):SET(0,46-I):SET(127,I):SET(127,46-I):
    NEXT
190 PRINT@1,"Score ";:PRINT@16,"Distance ";:
    PRINT@32,"Hi-score ";S$;" ";N$;:PRINT@961,"Energe ";
200 FORZ=15 TO 127:SET(Z,47):MUSIC=USR(Z):NEXT
210 A$(0)=CHR$(152)+CHR$(179)+CHR$(164):
    A$(1)=CHR$(137)+CHR$(179)+CHR$(134)
220 B$(0)=CHR$(157)+CHR$(131)+CHR$(174):
    B$(1)=CHR$(157)+CHR$(188)+CHR$(174)
230 C$(0)=CHR$(156)+CHR$(179)+CHR$(142):
    C$(1)=CHR$(141)+CHR$(179)+CHR$(172)
240 D$(0)=CHR$(157)+CHR$(179)+CHR$(174):
    D$(1)=CHR$(174)+CHR$(140)+CHR$(157)
250 E$(0)=CHR$(156)+CHR$(191)+CHR$(172):
    E$(1)=CHR$(152)+CHR$(143)+CHR$(164)
260 N=15425:FORZ=0TO50
270 N1=N+RND(1092):IFPEEK(N1)<>32THEN270
280 POKEN1,46:MUSIC=USR(89):R(Z)=N1:
    NEXT
290 RANDOM:J=RND(2):
    IFJ=1,PO=264+RND(20) ELSE
    PO=650+RND(20)
300 RANDOM:J=RND(3)
310 IFD,PRINT@OP,"     ";
320 IFJ=1,P=1:D=250:F$(0)=CHR$(129):F$(1)=CHR$(136):GOTO400
330 IFJ=2,P=2:D=150:
    F$(0)=CHR$(166)+CHR$(153):
    F$(1)=CHR$(153)+CHR$(166):GOTO400
340 RANDOM:J=RND(5)
350 F$(0)=E$(0):F$(1)=E$(1):M=100:D=50
360 IFJ=1,F$(0)=A$(0):F$(1)=A$(1):M=50
370 IFJ=2,F$(0)=B$(0):F$(1)=B$(1):M=60
```

```
380 IFJ=3,F$(0)=C$(0):F$(1)=C$(1):M=80
390 IFJ=4,F$(0)=D$(0):F$(1)=D$(1):M=90
400 FORV=0TO1:
        PRINT@PO,F$(0);:
                GOSUB830:PRINT@PO,F$(V);:
        GOSUB830:
    NEXT:
    OP=PO:POKE15904,43:PRINT@25,D;:PRINT@7,SC;
410 T=T+2:
    IFT>10,T=0:RESET(ST,47):MUSIC=USR(90):ST=ST-1:
    IFST=14,GOTO840
420 IFPEEK(14400)=8,PO=PO-RND(6)-62: GOTO510
430 IFPEEK(14400)=16,PO=PO+RND(6)+62:GOTO510
440 IFPEEK(14400)=32,PO=PO-RND(3):    GOTO510
450 IFPEEK(14400)=64,PO=PO+RND(3):    GOTO510
460 IFPEEK(14400)=40,PO=PO-RND(6)-64:GOTO510
470 IFPEEK(14400)=72,PO=PO+RND(6)-64:GOTO510
480 IFPEEK(14400)=48,PO=PO-RND(6)+64:GOTO510
490 IFPEEK(14400)=80,PO=PO+RND(6)+64
500 IFPEEK(14400)=2ANDD>90,D=D-10
510 X$=INKEY$:IFX$=" ",T=T+5:GOTO800
520 L=RND(50):L1=RND(50):L2=RND(50):
    POKER(L),32:POKER(L1),32:POKER(L2),32
530 RANDOM:J=RND(10):
    ONJ GOTO540,550,560,570,580,540,550,560,570,580
540 PO=PO-RND(8)-62: GOTO590
550 PO=PO-RND(4):     GOTO590
560 PO=PO+62+RND(8): GOTO590
570 PO=PO+RND(4):     GOTO590
580 IFP=1ORP=2,D=D+10
590 IFPO<65ORPO>955   THEN780
600 IFPO>123ANDPO<129THEN780
610 IFPO>187ANDPO<193THEN780
620 IFPO>251ANDPO<257THEN780
630 IFPO>315ANDPO<321THEN780
640 IFPO>379ANDPO<385THEN780
650 IFPO>443ANDPO<449THEN780
660 IFPO>507ANDPO<513THEN780
670 IFPO>571ANDPO<577THEN780
680 IFPO>635ANDPO<641THEN780
690 IFPO>699ANDPO<705THEN780
700 IFPO>763ANDPO<769THEN780
710 IFPO>827ANDPO<833THEN780
720 IFPO>891ANDPO<897THEN780
730 IFD>300              THEN780
740 IFP=1ANDD<200,J=2:     GOTO310
750 IFP=2ANDD<100,P=0:J=0:GOTO310
760 IFP=2ANDD>200THENJ=1:  GOTO310
770 PRINT@OP,"    ";:POKER(L),46:POKER(L1),46:
    POKER(L2),46:GOTO 400
780 PRINT@OP,"    ";:
              FORZ=1TO15:
                   PRINT@406,"Out of the fire range";:
              MUSIC=USR(Z*5):
        PRINT@406,STRING$(21,32);:
    NEXT:D=0:P=0:M=0:GOTO290
790 MUSIC=USR(5):RETURN
```

```
800 MUSIC=USR(90):SET(8,44):SET(118,44):SET(37,33):
    SET(89,33):SET(63,25):SET(66,25):
    MUSIC=USR(150):
    RESET(8,44):RESET(118,44):RESET(37,33):
    RESET(89,33):RESET(63,25):RESET(66,25)
810 IFOP>541ANDOP<547ANDD<100,SC=SC+M:
             FORI=1TO20:
                PRINT@542,STRING$(6,34+RND(4));:MUSIC=USR(I*5):
                PRINT@542,STRING$(8,32);:PRINT@479,M;:
             MUSIC=USR(I*10):
       PRINT@479,"     ";:
    NEXT:GOTO290
820 GOTO530
830 MUSIC=USR(2):RETURN
840 IFSC>VAL(S$),S$=STR$(SC)ELSE870
850 PRINT@532,"You set up a new record";:HL=1:N$=Y$
870 L=1:M$="Game Over (Hit Enter) to play again ":Q$=M$+M$
880 W$=MID$(Q$,L,36):PRINT@463,W$;:L=L+1:IFL=20,L=1
890 MUSIC=USR(L*30):IFPEEK(14400)=1,ST=127:SC=0:GOTO915
900 IFPEEK(14338)=64,FORI=1TO100:
    PRINT@462,"     Y o u  a r e  c h i c k e n ! ! !";:
    PRINT@462,STRING$(40,32);:MUSIC=USR(I*40):NEXT:CLS:END
910 GOTO880
915 IFHLTHEN160ELSE150
920 A$(0)=" "+CHR$(180)+CHR$(181)+CHR$(148):
    A$(1)=CHR$(168)+CHR$(169)+" ":
    A$(2)=" "+CHR$(152)+CHR$(137)+CHR$(144)
930 B$(0)=CHR$(184)+CHR$(189)+CHR$(144):
    B$(1)=STRING$(2,191)+CHR$(149):
    B$(2)=CHR$(184)+STRING$(2,141)+CHR$(173)+CHR$(144)
940 PRINT@465,;:INPUT"Please input your name";Y$:
    IFLEN(Y$)>15ORLEN(Y$)=0THEN940
950 CLS:PRINT@863,B$(0);:PRINT@927,B$(1);:PRINT@990,B$(2);
960 FORI=896TO921:
              PRINT@I,A$(0);:
                 PRINT@I+65,A$(1);:
                     FORY=1TO10:MUSIC=USR(Y*3):NEXT:
                 PRINT@I+65,A$(2);:
    NEXT:PRINT@921,"     ";:PRINT@985,"      ";
970 FORI=1TO8:
          PRINT:PRINT@992,CHR$(176):
             MUSIC=USR(I*7):FORY=1TO90STEP.5:NEXT:
    NEXT:RETURN
```

IE 80

個人資料檔案

電話、姓名紀錄庫

GENIE I

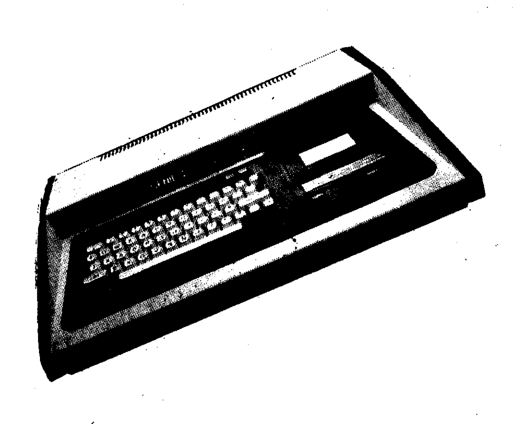
許多電腦軟件公司都有出產一些名爲MAIL－ING的程式軟件，爲一些須要儲存大量親朋戚友電話、地址的人提供一個方便。此類程式亦對工商業的商戶地址資料的記錄十分有用。

這程式是爲TRS－80及GENIE而設的，主要的功用是用來記錄他人的姓名及電話號碼。所記錄的個人資料數目爲200個，不過有時會因字串太長或其他個別情形，在未達200個資料之時，會有OUT MEHORY的情形。這時用家可先把資料輸送至錄音帶之中記存。跟著可從運行這程式（RUN），繼續另一個200個人資料庫。

本程式是考慮到GENIE的用家十分少數擁有磁碟機之故，所以錄音帶用家而設。

程式的各項功能如下：－

（1）輸入各項新資料。
（2）以一頁頁的方式去查看資料。
（3）用以查看個別的資料。
（4）用以更改或取消個別的某一項資料。
（5）把各項資料輸入錄音帶之中。
（6）把各項資料由錄音帶中讀回機中使用。



當輸入資料之時，其次序是姓名、電話號碼，及附錄，跟著是查看資料是否輸入錯誤，若果是有錯誤的話，可按下〝Y〞，重新輸入，如果沒有錯誤可按入〝N〞。如用家想返回MENU時，可在輸入姓名之時，按入〝1〞便可。

一頁頁去查看資料時，電腦會告訴你有幾多頁的資料（一頁資料時除外），跟著你可選擇由那一頁的資料開始。注意，如果在查看某資料時，發現沒有了尾的一段，不要慌，只要知道檔編號（由一頁頁的查看資料時便可知），用查看個別資料的功能，就把這個檔內的資料全印出來。

至於改錯及取消某一個檔案時，只要打入檔編號就可以了，改完之後，電腦立刻讓你看看你所新入的資料（用一頁頁的查看資料檔），當編號到時，一個〝>〞就在該檔案之前閃動，使你到注意到已改錯的檔案。

把資料送至卡式錄音機時，該部份爲GENIE II而設，因GENIE II唯一的CASSETTE NO. 1是外接的，如改用於GENIE I可把INPUT#－1或把1改爲2亦可，甚至不去理會INSTRUCTIONS，用CASSETTE NO. 1也很方便的。

由卡式帶中讀出資料之時，同上面的一樣，CA－SSTTE NO. 1是可隨意改變使用。

筆者提意用家兩盒錄音帶，一是用以SAVE PROGRAM，另一盒用以DATA記錄之用。

留意：每次把卡式帶中的資料讀回後再加以改動，又要SAVE PROGRAM之時，是要由頭S－AVE一次。

編輯話：此程式是磁帶版本，希望讀者把此程式更改在磁碟之用，並加上印字功能，歡迎讀者投稿。

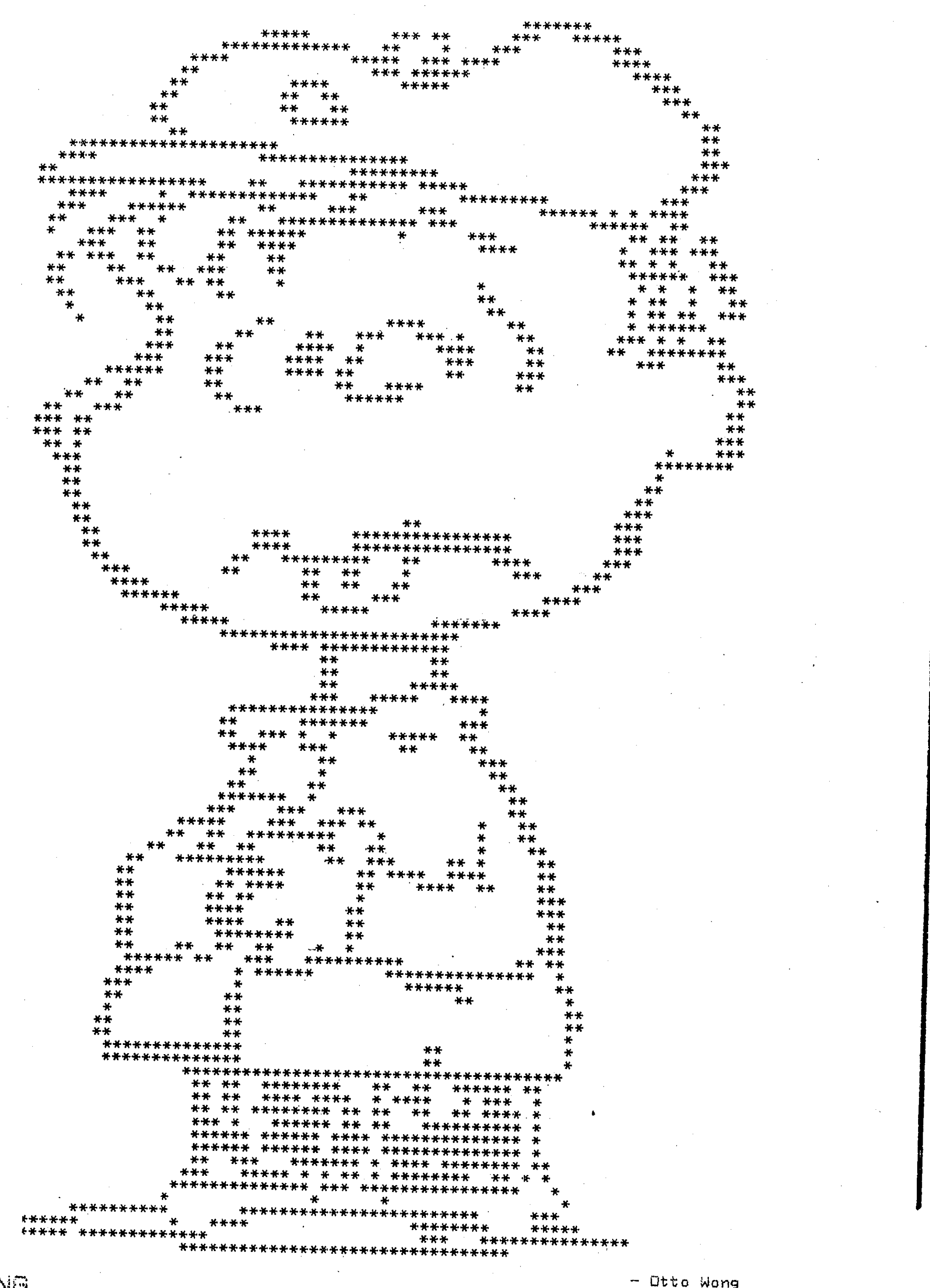
磁碟速度測試程式

GENIE 及 TRS－80

本程式為GENIE及TRS－80用家而設，程式十分簡單，一試即行。相信這對各磁碟之使用者有助。

```
10 PRINT"        *************************************************"
20 PRINT"                  TRS-80 MODEL I DISK TACKOMETER         "
30 PRINT"        *************************************************"
40 PRINT
50 CLS:R=-24320
60 PRINT"TACHOMETER IS LOADING"
70 FOR N=1 TO 79
80 READ D
90 POKE R,D
100 R=R+1
110 NEXT N
120 CLS
130 INPUT"DRIVE NUMBER";DN
140 CLS
150 IF DN<2 THEN 170
160 PRINT"DRIVE NUMBER TOO LARGE,TRY AGAIN":GOTO 130
170 DN=2^DN
180 POKE -24321,DN
190 DEF USR0=-24320
200 A=USR0(0)
210 M=-24574
220 RT=0:MAX=0:MIN=500
230 FOR N=1 TO 52
240 CO=PEEK(M)+256*PEEK(M+1)
250 TM=CO*4*9.75/1.77408/1000000
260 RPM=60/TM
270 RT=RT+RPM
280 PRINT RPM,
290 IF MAX<RPM THEN MAX=RPM
300 IF MIN>RPM THEN MIN=RPM
310 M=M+2
320 NEXT N
330 AV=RT/52
340 PRINT:PRINT"AVG RPM ="AV"   MAX RPM ="MAX"    MIN RPM="MIN
350 IF MAX>304.5 THEN PRINT "MAXIMUM SPEED TOO HIGH"
360 IF MIN<295.5 THEN PRINT "MINIMUM SPEED TOO LOW"
370 GOTO 130
380 DATA 243,058,255,160,050,225,055,062,003,050
390 DATA 236,055,001,000,128,205,096,000,058,236
400 DATA 055,203,071,032,249,058,255,160,050,225
410 DATA 055,001,000,160,017,000,000,058,236,055
420 DATA 203,079,040,249,058,236,055,019,203,079
430 DATA 032,248,000,058,236,055,019,203,079,040
440 DATA 248,123,002,003,122,002,003,058,255,160
450 DATA 050,225,055,062,106,185,032,212,201,000
```

GENIE 電腦圖案選 TRS-80



PROGRAM LISTING

– Otto Wong

```
5 REM   ********************
        ** SNOOPY FRIEND **
        * BY YIU WAH WONG *
        * COPYRIGHT  1982 *
        ********************
10 CLS:CLEAR 500:DEFINT A-Z:INPUT"SYMBOL";B$:CLS:PRINT"WORKING......";
:LPRINT CHR$(27)"8"
20 LPRINTCHR$(31)TAB(13)"SNOOPY FRIEND":LPRINTCHR$(30):LPRINT STRING$
(160,32):
```

```
30 X=ASC(B$)
40 READ A,B:IF A=999 GOTO 70
50 IF B=42 THEN B=X:GOTO 60
60 LPRINT STRING$(A,B);:GOTO 40
70 LPRINT STRING$(148,32)"- Otto Wong":GOTO 80
80 END
90 DATA 56,32,7,42,17,32
100 DATA 30,32,5,42,8,32,3,42,1,32,2,42,6,32,3,42,3,32,5,42,14,32
110 DATA 26,32,13,42,3,32,2,42,4,32,1,42,4,32,3,42,9,32,3,42,12,32
120 DATA 23,32,4,42,12,32,5,42,2,32,3,42,1,32,4,42,11,32,4,42,11,32
130 DATA 22,32,2,42,17,32,3,42,1,32,6,42,16,32,4,42,9,32
140 DATA 21,32,2,42,10,32,4,42,7,32,5,42,20,32,3,42,8,32
150 DATA 20,32,2,42,10,32,2,42,2,32,2,42,32,32,3,42,7,32
160 DATA 19,32,2,42,11,32,2,42,3,32,2,42,33,32,2,42,6,32
170 DATA 19,32,2,42,12,32,6,42,35,32,2,42,5,32
180 DATA 20,32,2,42,51,32,2,42,5,32
190 DATA 10,32,21,42,42,32,2,42,5,32
200 DATA 9,32,4,42,16,32,15,42,29,32,3,42,4,32
210 DATA 7,32,2,42,29,32,9,42,25,32,3,42,5,32
220 DATA 7,32,17,42,4,32,2,42,3,32,11,42,1,32,5,42,21,32,3,42,6,32
230 DATA 10,32,4,42,5,32,1,42,2,32,13,42,3,32,2,42,9,32,9,42,11,32,3,
42,8,32
240 DATA 9,32,3,42,4,32,6,42,7,32,2,42,5,32,3,42,6,32,3,42,9,32,6,42,1,
32,1,42,1,32,1,42,1,32,4,42,8,32
250 DATA 8,32,2,42,4,32,3,42,2,32,1,42,6,32,2,42,3,32,14,42,1,32,3,42,
13,32,6,42,2,32,2,42,8,32
260 DATA 8,32,1,42,3,32,3,42,2,32,2,42,6,32,2,42,1,32,6,42,9,32,1,42,6,
32,3,42,13,32,2,42,1,32,2,42,2,32,2,42,7,32
270 DATA 9,32,3,42,3,32,2,42,6,32,2,42,2,32,4,42,18,32,4,42,10,32,1,42,
2,32,3,42,1,32,3,42,6,32
280 DATA 8,32,2,42,1,32,3,42,2,32,2,42,5,32,2,42,4,32,2,42,33,32,2,42,1
,32,1,42,1,32,1,42,3,32,2,42,5,32
290 DATA 7,32,2,42,4,32,2,42,3,32,2,42,2,32,3,42,4,32,2,42,34,32,6,42,
2,32,3,42,4,32
300 DATA 7,32,2,42,5,32,3,42,3,32,2,42,1,32,2,42,5,32,1,42,19,32,1,42,
15,32,1,42,1,32,1,42,2,32,1,42,2,32,2,42,5,32
310 DATA 7,32,2,42,6,32,2,42,6,32,2,42,24,32,2,42,13,32,1,42,1,32,2,42,
2,32,1,42,3,32,2,42,4,32
320 DATA 8,32,1,42,7,32,2,42,32,32,2,42,12,32,1,42,1,32,2,42,1,32,2,42,
2,32,3,42,4,32
330 DATA 9,32,1,42,7,32,2,42,8,32,2,42,11,32,4,42,8,32,2,42,10,32,1,42,
1,32,6,42,8,32
340 DATA 17,32,2,42,6,32,2,42,5,32,2,42,3,32,3,42,3,32,3,42,1,32,1,42,
5,32,2,42,8,32,3,42,1,32,1,42,1,32,1,42,2,32,2,42,6,32
350 DATA 16,32,3,42,4,32,2,42,6,32,4,42,2,32,1,42,7,32,4,42,5,32,2,42,
6,32,2,42,2,32,8,42,6,32
360 DATA 15,32,3,42,4,32,3,42,5,32,4,42,2,32,2,42,8,32,3,42,5,32,2,42,
9,32,3,42,5,32,2,42,5,32
370 DATA 12,32,6,42,4,32,2,42,6,32,4,42,1,32,2,42,9,32,2,42,5,32,3,42,
17,32,3,42,4,32
380 DATA 10,32,2,42,2,32,2,42,6,32,2,42,11,32,2,42,3,32,4,42,9,32,2,
42,20,32,2,42,3,32
390 DATA 8,32,2,42,3,32,2,42,8,32,2,42,11,32,6,42,33,32,2,42,3,32
400 DATA 6,32,2,42,3,32,3,42,11,32,3,42,46,32,2,42,4,32
410 DATA 5,32,3,42,1,32,2,42,63,32,2,42,4,32
420 DATA 5,32,3,42,1,32,2,42,62,32,3,42,4,32
430 DATA 6,32,2,42,1,32,1,42,58,32,1,42,4,32,3,42,4,32
440 DATA 7,32,3,42,57,32,8,42,5,32
450 DATA 8,32,2,42,57,32,1,42,12,32
460 DATA 8,32,2,42,56,32,2,42,12,32
470 DATA 8,32,2,42,55,32,2,42,13,32
480 DATA 9,32,2,42,53,32,3,42,13,32
490 DATA 9,32,2,42,31,32,2,42,19,32,3,42,14,32
500 DATA 10,32,2,42,15,32,4,42,6,32,16,42,10,32,3,42,14,32
510 DATA 10,32,2,42,15,32,4,42,6,32,16,42,10,32,3,42,14,32
520 DATA 11,32,2,42,12,32,2,42,3,32,9,42,3,32,2,42,7,32,4,42,7,32,3,42,
15,32
530 DATA 12,32,3,42,9,32,2,42,6,32,2,42,2,32,2,42,4,32,1,42,10,32,3,42,
5,32,2,42,17,32
```

```
540 DATA 13,32,4,42,15,32,2,42,2,32,2,42,3,32,2,42,16,32,3,42,18,32
550 DATA 14,32,6,42,12,32,2,42,5,32,3,42,14,32,4,42,20,32
560 DATA 18,32,5,42,11,32,5,42,14,32,4,42,23,32,20,32,5,42,20,32,7,42,
27,32
570 DATA 25,32,24,42,31,32
580 DATA 30,32,4,42,1,32,13,42,32,32
590 DATA 35,32,2,42,9,32,2,42,32,32
600 DATA 35,32,2,42,9,32,2,42,32,32
610 DATA 35,32,2,42,7,32,5,42,31,32
620 DATA 34,32,3,42,3,32,5,42,3,32,4,42,28,32
630 DATA 26,32,15,42,10,32,1,42,28,32
640 DATA 25,32,2,42,6,32,7,42,9,32,3,42,28,32
650 DATA 25,32,2,42,2,32,3,42,1,32,1,42,2,32,1,42,5,32,5,42,2,32,2,42,
29,32
660 DATA 26,32,4,42,3,32,3,42,7,32,2,42,5,32,2,42,28,32
670 DATA 28,32,1,42,6,32,2,42,14,32,3,42,26,32
680 DATA 27,32,2,42,6,32,1,42,16,32,2,42,26,32
690 DATA 26,32,2,42,6,32,2,42,17,32,2,42,25,32
700 DATA 25,32,7,42,2,32,1,42,19,32,2,42,24,32
710 DATA 24,32,3,42,4,32,3,42,3,32,3,42,14,32,2,42,24,32
720 DATA 21,32,5,42,4,32,3,42,2,32,3,42,1,32,2,42,10,32,1,42,3,32,2,
42,23,32
730 DATA 20,32,2,42,2,32,2,42,2,32,9,42,4,32,1,42,9,32,1,42,3,32,2,42,
23,32
740 DATA 18,32,2,42,3,32,2,42,2,32,2,42,6,32,2,42,3,32,2,42,9,32,1,42,
4,32,2,42,22,32
750 DATA 16,32,2,42,3,32,9,42,6,32,2,42,2,32,3,42,5,32,2,42,1,32,1,42,
5,32,2,42,21,32
760 DATA 15,32,2,42,9,32,6,42,7,32,2,42,1,32,4,42,2,32,4,42,5,32,2,42,
21,32
770 DATA 15,32,2,42,8,32,2,42,1,32,4,42,7,32,2,42,4,32,4,42,2,32,2,42,
4,32,2,42,21,32
780 DATA 15,32,2,42,7,32,2,42,1,32,2,42,10,32,1,42,17,32,3,42,20,32
790 DATA 15,32,2,42,7,32,4,42,10,32,2,42,17,32,3,42,20,32
800 DATA 15,32,2,42,7,32,4,42,3,32,2,42,5,32,2,42,18,32,2,42,20,32
810 DATA 15,32,2,42,8,32,8,42,5,32,2,42,18,32,2,42,20,32
820 DATA 15,32,2,42,4,32,2,42,2,32,2,42,2,32,2,42,4,32,1,42,2,32,1,42,
18,32,3,42,20,32
830 DATA 16,32,6,42,1,32,2,42,3,32,3,42,3,32,10,42,11,32,2,42,1,32,2,42
,20,32
840 DATA 15,32,4,42,8,32,1,42,1,32,6,42,7,32,15,42,2,32,1,42,20,32
850 DATA 14,32,3,42,10,32,1,42,16,32,6,42,9,32,2,42,19,32
860 DATA 14,32,2,42,10,32,2,42,21,32,2,42,9,32,1,42,19,32
870 DATA 14,32,1,42,11,32,2,42,32,32,2,42,18,32
880 DATA 13,32,2,42,11,32,2,42,32,32,2,42,18,32
890 DATA 13,32,2,42,11,32,2,42,32,32,1,42,19,32
900 DATA 14,32,14,42,18,32,2,42,12,32,1,42,19,32
910 DATA 14,32,14,42,18,32,2,42,12,32,1,42,19,32
920 DATA 22,32,38,42,20,32
930 DATA 23,32,2,42,1,32,2,42,2,32,8,42,3,32,2,42,2,32,2,42,2,32,6,42,
1,32,2,42,22,32
940 DATA 23,32,2,42,1,32,2,42,2,32,4,42,1,32,4,42,2,32,1,42,1,32,4,42,
3,32,1,42,1,32,3,42,2,32,1,42,22,32,2,32
950 DATA 21,32,2,42,1,32,2,42,1,32,2,42,6,42,1,32,2,42,1,32,2,42,2,32,
2,42,2,32,2,42,1,32,4,42,1,32,1,42,22,32.
960 DATA 23,32,3,42,1,32,1,42,3,32,6,42,1,32,2,42,1,32,2,42,3,32,10,42,
1,32,1,42,22,32
970 DATA 23,32,6,42,1,32,6,42,1,32,4,42,1,32,14,42,1,32,1,42,22,32
980 DATA 23,32,6,42,1,32,6,42,1,32,4,42,1,32,14,42,1,32,1,42,22,32
990 DATA 23,32,2,42,2,32,3,42,3,32,7,42,1,32,1,42,1,32,4,42,1,32,8,42,
1,32,2,42,21,32
1000 DATA 22,32,3,42,3,32,5,42,1,32,1,42,1,32,1,42,1,32,2,42,1,32,1,42,
1,32,8,42,2,32,2,42,1,32,1,42,1,32,1,42,21,32
1010 DATA 21,32,14,42,1,32,3,42,1,32,16,42,3,32,1,42,20,32
1020 DATA 20,32,1,42,14,32,1,42,6,32,1,42,17,32,1,42,19,32
1030 DATA 11,32,10,42,7,32,24,42,5,32,3,42,20,32
1040 DATA 6,32,6,42,9,32,1,42,3,32,4,42,16,32,8,42,4,32,5,42,17,32
1050 DATA 5,32,7,42,1,32,13,42,21,32,3,42,3,32,15,42,12,32
1060 DATA 23,32,33,42,24,32
1070 DATA 999,999
```

開閉遊戲鬥智程式

ON/OFF GAME

602P, PC-1500 版本

以人的智慧和電腦鬥智，若能勝電腦可謂一流，但有時則是運氣。此類遊戲都是由電腦隨意產生一些數字，而玩者需小心選擇要關掉的燈，直至一方不能關掉燈泡為止。

此遊戲有PC－1500及FX－602P的版本，而FX602P版本有兩個不同的程式，一為顯示屏的程式，另一為印字機的程式。程式名為：—"ON／OFF GAME"

602P 顯示屏版本

首先介紹顯示屏玩的版本。遊戲大約可分為三部份。第一部份是用以決定由誰先行第一步。當按下Pφ後，有"WHO？"顯示。如果由機玩先，即按"φ""EXE"。如果由你玩先即按下"EXE"便可。

第二部份是會有兩個隨機數由1至6。例如4，6，你便可以關掉總數為10的燈泡，如2及8，5及5，1及9等等。而玩者只有三次輸入機會，而沒有燈可關掉時即有以下的字樣"ON CHOICE"並定輸贏。

第三部份是當602P關燈之後，有二份之一的機會由機去把其中一燈泡重新開回。當時有"〔O－N〕〔A〕"的字樣，而A為所亮的燈的位置。

遊戲直至沒有燈能熄滅為止。而FX－602P在思考時有問號顯示增加趣味性。當你忘掉那燈已熄掉時，你可在隨機數顯示之後，按下"EXE"圖像便再顯示出來。燈號為1至9號從左至右。以0代表亮燈，*代表關上的燈。當輸入為1個數字，而又符合關燈條件，便不會去追問第二個數字。如：隨機數為4，6，可選5號這燈泡。

圖一為本程式，輸入次序為：P6，P7，P8，P5，Pφ。

```
PROGRAM LIST
M00-19,F-1F 512steps

*** P0
4 Min00
LBL0
"ON      "
"OFF "
DSZ GOTO0
MAC 1 Min10
9 Min00
LBL0
MR00 IND Min00
DSZ GOTO0
π
"WHO ?"
HLT x=0
AC Min11
9 Min12
")"
LBL2
IND MR12 M+1F x=0
GOTO3
";0"
1 M-12
MR12 x=0 GOTO4
GOTO2
LBL3
";*"
1 M-12
MR12 x=0 GOTO4
GOTO2
LBL3
"OVER  "
PAUSE
GSBP6
LBL4
";("
PAUSE
";"
1 Min10
MR1F x=0 GOTO3
MR11 x=0 GOTO7
MR11 - 2 = x=0 GOTO5
MR11 - 3 = x=0
GSBP7
MR10 x=0 GOTO2
GSBP5
Min13
GSBP5
Min14 + MR13 = MinF
4 Min19
GSBP8
GOTO5
LBL0
1 M+19 Min11 M+11
9 Min12
")"
GOTO2
LBL5
1 M-19
MR19 x=0
GSBP6
AC Min16
"_  AR13 , AR14  - "
HLT Min15 x=0 GOTO0
x=F GOTO9
Min16 - 10 = x≥0
GOTO5
MR15 x 2 = x=F GOTO9
AC
"?"
HLT Min16 - 10 = x≥0
GOTO5
GOTO9
LBL0
0 Min19
"NO CHOICE"
PAUSE
GSBP6
LBL9
MR00 x=0 GOTO0
IND MR15 x=0 GOTO5
IND MR16 x=0 GOTO5
MR15 + MR16 = x=F
GOTO6
GOTO5
LBL6
"OFF  AR15 , AR16 "
PAUSE
0 IND Min15 IND
Min16 Min11 Min10
MinF
")"
9 Min12 GOTO2
LBL9
";?"
DSZ GOTO8
"Oh!!!"
PAUSE
GSBP6
LBL7
GSBP5
Min13
GSBP5
Min14 + MR13 = MinF
"(   AR13 , AR14  )
"
PAUSE
"?"
9 Min00
";?"
LBL8
IND MR00 x=0 GOTO9
MRF - MR00 = ABS
Min17 - 10 = x≥0
GOTO9
IND MR17 x=0 GOTO9
IND MR00 + MR17 =
x=F GOTO1
GOTO9
LBL5
3 Min11
")"
GOTO2
LBL1
"Ah!! "
PAUSE
"OFF  AR17 , AR00 "
PAUSE
1 Min11
0 IND Min00 IND
Min17 Min19 MinF
9 Min12
RAN#
RAN# x 1.6 = INT x=0
GOTO5
")"
GOTO2          ...378steps

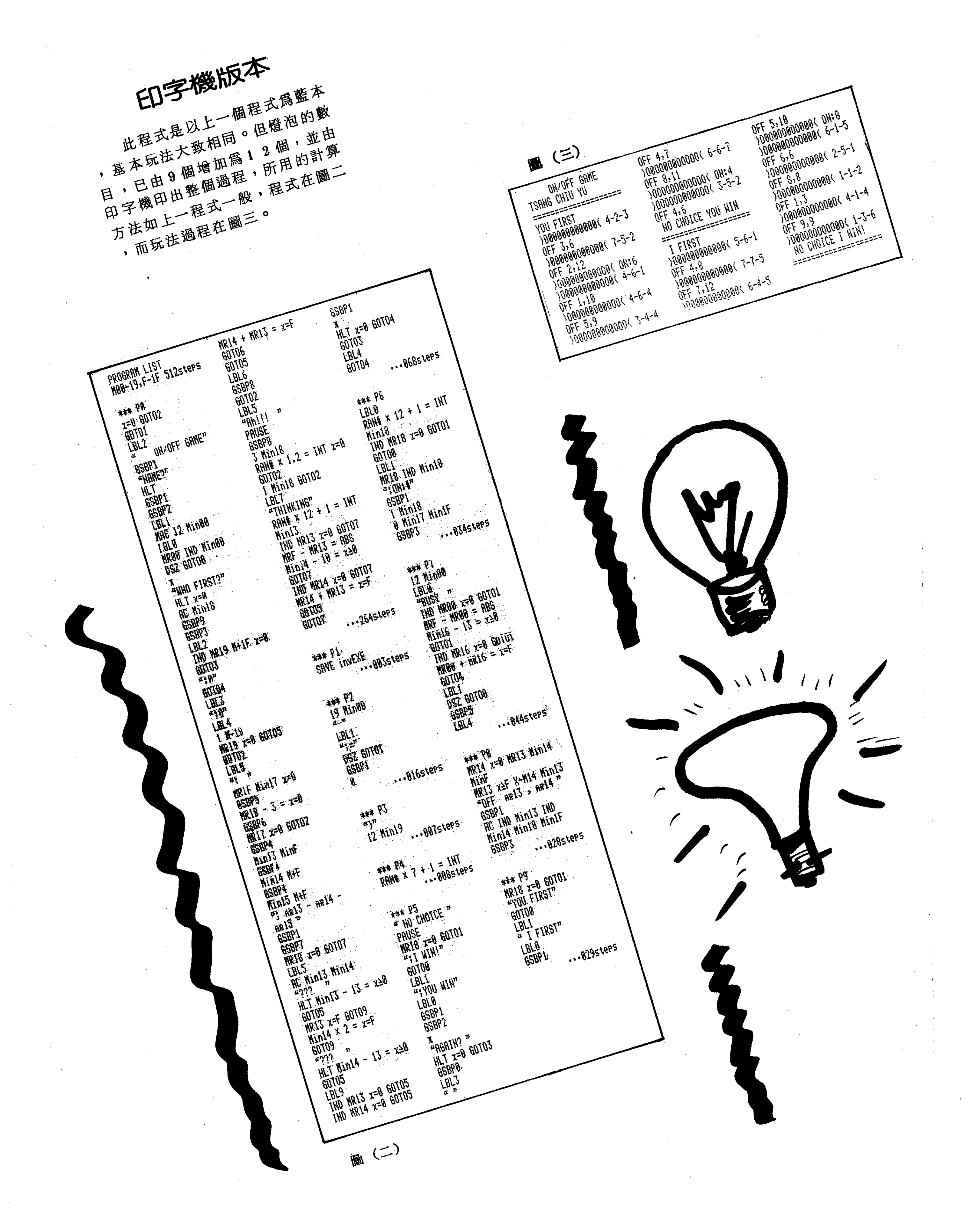
*** P6
MR19 x=0 GOTO3
MR1F x=0 GOTO2
LBL1
"YOU WIN!"
PAUSE GOTO1
LBL2
MR19 x≥0 GOTO1
LBL3
"I WIN! "
PAUSE GOTO3
               ...036steps

*** P7
"( ON ) "
LBL1
RAN# x 9 + 1 = INT
Min10
IND MR10 x=0 GOTO2
GOTO1
LBL2
MR10 IND Min10
";( AR10 )"
PAUSE
1 Min11 Min10 M-10
9 Min12
")"            ...044steps

*** P8
9 Min00 GOTO2
LBL1
";"
DSZ GOTO2
GOTO4
LBL2
IND MR00 x=0 GOTO1
MRF - MR00 = ABS
Min17 - 10 = x≥0
GOTO1
IND MR17 x=0 GOTO1
MR17 + IND MR00 =
x=F GOTO4
GOTO1
LBL4           ...041steps

*** P5
RAN# x 6 + 1 = INT
               ...008steps
```

圖（一）



印字機版本

此程式是以上一個程式爲藍本，基本玩法大致相同。但燈泡的數目，已由9個增加爲12個，並由印字機印出整個過程，所用的計算方法如上一程式一般，程式在圖二，而玩法過程在圖三。

```
PROGRAM LIST
M00-19,F-1F 512steps

 *** PA
 x=0 GOTO2
 GOTO1
 LBL2
 "    ON/OFF GAME"
  GSBP1
  "NAME?"
  HLT
  GSBP1
  GSBP2
  LBL1
  MAC 12 Min00
  LBL0
  MR00 IND Min00
  DSZ GOTO0
   x
   "WHO FIRST?"
   HLT x=0
   AC Min18
   GSBP9
   GSBP3
   LBL2
   IND MR19 M+1F x=0
   GOTO3
   ";0"
   GOTO4
   LBL3
   ";1"
   LBL4
   1 M-19
   MR19 x=0 GOTO5
   GOTO2
   LBL5
   ";   "
   MR1F Min17 x=0
   GSBP5
   MR18 - 3 = x=0
   GSBP6
   MR17 x=0 GOTO2
   GSBP4
   Min13 MinF
   GSBP4
   Min14 M+F
   GSBP4
   Min15 M+F
   "; AR13 - AR14 -
    AR15 "
   GSBP1
   GSBP7
   MR18 x=0 GOTO7
   LBL5
   AC Min13 Min14
   "???    "
   HLT Min13 - 13 = x≥0
   GOTO5
   MR13 x=F GOTO9
   Min14 x 2 = x=F
   GOTO9
   "???    "
   HLT Min14 - 13 = x≥0
   GOTO5
   LBL9
   IND MR13 x=0 GOTO5
   IND MR14 x=0 GOTO5

MR14 + MR13 = x=F
GOTO6
GOTO5
LBL6
GSBP8
GOTO2
LBL5
"Ah!!!  "
PAUSE
GSBP8
3 Min18
RAN# x 1.2 = INT x=0
GOTO2
1 Min18 GOTO2
LBL7
"THINKING"
RAN# x 12 + 1 = INT
Min13
IND MR13 x=0 GOTO7
MRF - MR13 = ABS
Min14 - 10 = x≥0
GOTO7
IND MR14 x=0 GOTO7
MR14 + MR13 = x=F
GOTO5
GOTO7        ...264steps

 *** P1
 SAVE invEXE
          ...003steps

 *** P2
 19 Min00
 "_"
 LBL1
 ";="
 DSZ GOTO1
 GSBP1
 0          ...016steps

  *** P3
  ")"
  12 Min19    ...007steps

  *** P4
  RAN# x 7 + 1 = INT
             ...008steps

  *** P5
  " NO CHOICE "
  PAUSE
  MR18 x=0 GOTO1
  ";I WIN!"
  GOTO0
  LBL1
  ";YOU WIN"
  LBL0
  GSBP1
  GSBP2
   x
   "AGAIN? "
   HLT x=0 GOTO3
   GSBP0
   LBL3
   " "

GSBP1
x
HLT x=0 GOTO4
GOTO3
LBL4
GOTO4        ...068steps

 *** P6
 LBL0
 RAN# x 12 + 1 = INT
 Min18
 IND MR18 x=0 GOTO1
 GOTO0
 LBL1
 MR18 IND Min18
 ";ON:#"
 GSBP1
 1 Min18
 0 Min17 Min1F
 GSBP3       ...034steps

  *** P7
  12 Min00
  LBL0
  "BUSY  "
  IND MR00 x=0 GOTO1
  MRF - MR00 = ABS
  Min16 - 13 = x≥0
  GOTO1
  IND MR16 x=0 GOTO1
  MR00 + MR16 = x=F
  GOTO4
  LBL1
  DSZ GOTO0
  GSBP5
  LBL4         ...044steps

   *** P8
   MR14 x=0 MR13 Min14
   MinF
   MR13 x≥F X↔M14 Min13
   "OFF  AR13 , AR14 "
   GSBP1
   AC IND Min13 IND
   Min14 Min18 Min1F
   GSBP3       ...028steps

    *** P9
    MR18 x=0 GOTO1
    "YOU FIRST"
    GOTO0
    LBL1
    " I FIRST"
    LBL0
    GSBP1       ...029steps
```

圖（二）

圖（三）

```
   ON/OFF GAME
TSANG CHIU YU
==================
YOU FIRST
)000000000000( 4-2-3
OFF 3,6
)000000000000( 7-5-2
OFF 2,12
)000000000000( ON:6
)000000000000( 4-6-1
OFF 1,10
)000000000000( 4-6-4
OFF 5,9
)000000000000( 3-4-4

OFF 4,7
)000000000000( 6-6-7
OFF 8,11
)000000000000( ON:4
)000000000000( 3-5-2
OFF 4,6
 NO CHOICE YOU WIN
==================
 I FIRST
)000000000000( 5-6-1
OFF 4,8
)000000000000( 7-7-5
OFF 7,12
)000000000000( 6-4-5

OFF 5,10
)000000000000( ON:8
)000000000000( 6-1-5
OFF 6,6
)000000000000( 2-5-1
OFF 8,8
)000000000000( 1-1-2
OFF 1,3
)000000000000( 4-1-4
OFF 9,9
)000000000000( 1-3-6
 NO CHOICE I WIN!
==================
```

(ON): 3

PC-1500

玩者可由顯示屏上看到燈泡的形狀，及其變數，增加趣味。本程式盡量利用ＰＣ－１５００的獨有顯示功能及其聲音效果，玩者可和電腦作一個比賽。本程式詳列於圖四。

WHO FIRST?

3 5 2 ?

OFF 2 10

```
  1:REM **********
  2:REM * ON OFF *
  3:REM *  GAME  *
  4:REM *BY  IVAN*
  5:REM **********
 10:DIM A(14),T$(3
    )*20:RANDOM :
    POKE &79FF,0
 20:T$(0)="000E116
    D6D110E00":T$(
    1)="000E116161
    110E00"
 40:FOR X=1TO 63
 50:GCURSOR (X+1)
 60:WAIT 0:PRINT "
    ON"
 65:BEEP 1,25,50
 70:GCURSOR (144-X
    )
 80:WAIT 0:PRINT "
    OFF"
 85:BEEP 1,52,30
 90:NEXT X
 91:FOR U=1TO 3
 92:BEEP 2,85,20
 93:BEEP 1,52,30
 94:BEEP 1,101,90
 95:BEEP 3,63,40
 96:NEXT U
100:FOR X=65TO 155
110:GCURSOR X
120:WAIT 3:PRINT "
    ON OFF"
125:BEEP 1,15,10
130:NEXT X
140:FOR Z=1TO 13
150:A(Z)=Z
160:NEXT Z
170:GCURSOR 43:
    BEEP 1,10,20:
    PRINT " WHO FI
    RST? ":P$=
    INKEY$ :IF P$=
    ""GOTO 170
180:BEEP 1,100,15
190:IF P$="M"GOTO
    320
200:IF P$="Y"GOTO
    500
210:GOTO 170
300:GCURSOR 0
305:FOR X=1TO 13
310:WAIT 0:IF A(X)
    =0GPRINT T$(1)
    ;:BEEP 1,50,20
    :GOTO 315
311:WAIT 0:GPRINT
    T$(0);:BEEP 1,
    70,20
315:NEXT X:PRINT "
              ":
    BEEP 5,100,200
    :RETURN
320:GOSUB 300
321:GCURSOR 108:
    WAIT 70:PRINT
    "YOUR.T.":BEEP
    3,5,50:C=RND 6
    :D=RND 6:E=RND
    6:F=C+D+E
325:FOR X=1TO 14
330:IF A(X)=0GOTO
    340
331:IF F=A(X)GOTO
    350
335:N=ABS (F-A(X))
336:IF N>=14GOTO 3
    49
340:IF A(N)=0GOTO
    349
345:IF F<>(N+A(X))
    GOTO 349
346:IF F=(N+A(X))
    GOTO 350
349:NEXT X
350:IF X=14GOTO 10
    00
360:Q=0:W=0:
    GCURSOR 103:
    WAIT 0:PRINT C
    ;D;E;"  ";:
    BEEP 4,70,60:
    INPUT Q:BEEP 1
    ,80,40
370:IF Q>=14GOTO 3
    60
371:IF Q=(F/2)LET
    W=Q
372:IF Q=WGOTO 390
373:IF Q=FGOTO 390
374:GCURSOR 103:
    WAIT 0:PRINT C
    ;D;E;"  ";:
    INPUT W:BEEP 1
    ,80,40
375:IF W>=14GOTO 3
    60
380:IF A(Q)=0GOTO
    360
385:IF A(W)=0GOTO
    360
390:IF F<>(A(Q)+A(
    W))GOTO 360
400:GCURSOR 108:
    WAIT 70:PRINT
    "OFF";Q;W:BEEP
    2,150,100
405:A(Q)=0:A(W)=0
500:GOSUB 300
530:GCURSOR 108:
    WAIT 70:PRINT
    "MY T. ":BEEP
    3,5,50:C=RND 6
    :D=RND 6:E=RND
    6:F=C+D+E
535:FOR X=1TO 14
540:IF A(X)=0GOTO
    560
541:IF F=A(X)GOTO
    565
545:N=ABS (F-A(X))
    :IF N>=14GOTO
    560
550:IF A(N)=0GOTO
    560
555:IF F<>(N+A(X))
    GOTO 560
556:IF F=(N+A(X))
    GOTO 565
560:NEXT X
565:IF X=14GOTO 20
    00
570:GCURSOR 108:
    WAIT 70:PRINT
    C;D;E:BEEP 2,7
    0,60
575:Q=0:W=0:Q=RND
    13
580:IF A(Q)=0GOTO
    575
581:IF F=A(Q)GOTO
    595
585:W=ABS (F-A(Q))
    :IF W>=14GOTO
    575
586:IF A(W)=0GOTO
    575
590:IF W>=14GOTO 5
    75
595:IF F<>(A(Q)+A(
    W))GOTO 575
600:A(Q)=0:A(W)=0:
    GCURSOR 108:
    WAIT 50:PRINT
    "OFF";Q;W:BEEF
    2,100,100
605:GCURSOR 0
610:FOR X=1TO 13
615:WAIT 0:IF A(X)
    =0GPRINT T$(1)
    ;:BEEP 1,50,20
    :GOTO 625
620:WAIT 0:GPRINT
    T$(0);:BEEP 1,
    70,20
625:NEXT X:PRINT "
              ":
    BEEP 5,100,20
630:O=RND 2:IF O=1
    GOTO 640
635:GOTO 321
640:Y=RND 13
643:IF A(Y)<>0GOTO
    640
645:IF A(Y)=0LET A
    (Y)=Y
647:GCURSOR 108:
    WAIT 70:PRINT
    "(ON):";Y:BEEP
    2,150,70:BEEP
    2,120,30:BEEP
    2,60,20:GOTO 3
    20
1000:GCURSOR 108:
     WAIT 80:
     PRINT C;D;E:
     BEEP 3,80,40
     :CLS
1001:FOR X=15TO 7
     7
1002:WAIT 0:
     GCURSOR X:
     GPRINT "55":
     GCURSOR (156
     -X):GPRINT "
     55":BEEP 1,9
     0,10:BEEP 1,
     40,20
1003:NEXT X
1004:FOR X=77TO 9
     9
1005:WAIT 0:
     GCURSOR X:
     GPRINT "7F":
     GCURSOR (155
     -X):GPRINT "
     7F":BEEP 1,5
     5,30
1006:NEXT X
1007:GCURSOR 59:
     GPRINT "0000
     ";:PRINT "I
     WIN!":GOSUB
     3000:PAUSE :
     END
2000:GCURSOR 108:
     WAIT 70:
     PRINT C;D;E:
     BEEP 3,50,50
     :WAIT 0:CLS
2001:FOR M=2TO 54
2002:GCURSOR M:
     GPRINT "00";
     :WAIT 1:
     PRINT "YOU W
     IN!"
2004:GCURSOR (108
     -M):WAIT 0:
     PRINT "YOU W
     IN!"
2007:BEEP 2,30,20
     :BEEP 1,50,3
     0:NEXT M
2008:GOSUB 3000:
     PAUSE :END
3000:FOR L=1TO 2
3001:BEEP 3,90,50
3002:BEEP 1,70,50
3003:BEEP 1,150,9
     0
3004:BEEP 1,150,1
     00
3005:BEEP 2,60,60
3006:BEEP 1,200,2
     00
3007:BEEP 2,150,1
     00
3008:NEXT L
3009:RETURN
```

圖（四）

STAR WAR

PC-1500

新紀元９４１年，月球殖民地宣告獨立，改稱帝國，脫離地球聯邦政府的統治，殖民地司令稱帝，自號爲希特拉三世，而且向地球聯邦軍正式宣戰，由於殖民地的主要居民都是科學家和技師，所以軍事力量非常強大，他們設計了一種新式的戰鬥機——ＦＯＸ９４１——作爲攻擊地球的主力：而地球聯邦軍，亦發展了一種名爲——ＰＣ１５——的反擊系統，但需要大量的炮手，於是ＰＣ１５訓練炮手計劃亦告展開了。

在訓練艙裏，一切的設備都是模仿眞正的機艙而製造的，包括螢幕顯示和三個控制按鍵，每次出擊配給９９個燃料單位，一開始時螢幕在右手邊會出現剩餘的燃料單位，螢幕中央出現〔　〕一個符號，此乃瞄準器是也，而敵機——ＦＯ－Ｘ　９４１－則出現於附近，形如〔Ｏ〕。爲了鼓勵士氣，在作戰之前，會奏出ＳＴＡＲ　ＷＡＲ主題曲之一小部份！以壯聲威。

註：（１）倘若一擊不中，會扣５個燃料單位。

（２）擊中後，會加上１０個燃料單位；而得分是３０，４０或５０。

最後ＧＡＭＥ　ＯＶＥＲ，會顯示最高得分者之名字及其分數，現在最高記錄爲７５０ＰＯＩＮＴ－Ｓ。

```
LIST STAR WAR
 10:"A"RANDOM :
    WAIT 0:Z=53
 20:S=0:T=100:R=12
    7:A$="0808":B$
    ="0000"
 30:PRINT "    >> S
    TAR WARS <<":
    RESTORE
 50:FOR A=1TO 16:
    READ B,C:BEEP
    1,B,C:NEXT A:
    CLS
 60:CLS :X=18+RND
    82
 70:E$=INKEY$
 80:IF E$="" THEN
    205
 90:IF E$="3"LET Z
    =Z+2:IF Z>109
    LET Z=109
100:IF E$="1"LET Z
    =Z-2:IF Z<0LET
    Z=0
110:X=X-2+RND 3
120:GCURSOR X:
    GPRINT "000000
    3E491C361C493E
    000000":
    GCURSOR Z:
    GPRINT "00007F
    410000"
123:GCURSOR Z+13:
    GPRINT "000041
    7F0000"
125:CURSOR 21
    PRINT USING "#
    ###";T
130:T=T-0.5:IF T<0
    THEN 150
140:GOTO 70
150:FOR D=10TO 100
    :BEEP 1,D,10:
    NEXT D
160:IF H>=STHEN 18
    0
170:BEEP 10:H=S:
    PAUSE "YOU ARE
    TOP!  ";H:
    PAUSE :INPUT "
    YOUR NAME ? ";
    H$:GOTO 10
180:BEEP 1:PAUSE "
    TOP POINT ";H:
    PAUSE "BY ";H$
190:WAIT :BEEP 1:
    PRINT "GAME OV
    ER ";S:END
205:P=POINT (Z+9):
    FOR E=10TO 30:
    BEEP 1,E,10
207:GCURSOR Z+1:
    GPRINT E:
    GCURSOR Z+17:
    GPRINT E
209:NEXT E
220:IF P=54THEN 23
    0
225:CLS :T=T-5:T=T
    *(1+SGN T)/2:
    GOTO 110
230:FOR F=200TO 23
    0STEP 4
240:GCURSOR X+2:
    GPRINT "084922
    085D08224908"
245:GCURSOR Z+19:
    GPRINT "007020
    70007000107010
    "
250:BEEP 1,F,10
260:NEXT F
270:G=10*(RND 3+2)
    :T=T+10
280:S=S+G:PAUSE "
    POINTS";G:Z=53
    :GOTO 60
300:DATA 100,500,6
    5,800,75,72,80
    ,70,90,60,47,1
    000,65,700
310:DATA 75,72,80,
    70,90,60,47,10
    00,65,700
320:DATA 75,72,80,
    70,75,72,90,96
    0
STATUS 1         941
```

FU-PRINTING

PC-1500

FU-PRINTING (PC-1500) with CE150

很多人都會將CE-150彩色編印機去做一些美妙圖案，用來作一些擺設，但也有些用作書簽。

筆者花了差不多一個星期才完成那福字編印，CE-150用家可以用作聖誕咭。或新年咭寄給朋友們。

此程序使用法相當簡單，按R-UN便可。



Sample of FU



```
1:REM  FU-PRINTI
   NG
10:"A":GRAPH :X1=
   0
20:READ X2,X:X2=X
   2+50:IF X1>X2
   THEN GLCURSOR
   (0,-2):SORGN
30:IF X2=0AND X=0
   THEN END
40:X1=X2+X:LINE (
   X2,0)-(X1,0):
   LINE (X2,-1)-(
   X1,-1):GOTO 20
50:DATA 98,5,97,8
   ,95,12,93,15,9
   1,19,37,1,88,2
   3,36,3,83,29
60:DATA 36,6,78,3
   5,34,9,46,5,69
   ,45,33,20,70,4
   5,32,22,71,44
70:DATA 32,23,72,
   43,32,24,73,41
   ,32,25,74,39,3
   2,26,75,36
80:DATA 32,27,75,
   33,33,26,75,29
   ,34,26,74,2,78
   ,22,35,25,72,1
   ,79,15
90:DATA 37,22,101
   ,9,38,21,68,2,
   95,19,40,18,67
   ,4,91,25,43,12
   ,66,5,87,30,46
   ,6,65,7
100:DATA 82,36,64,
   8,79,40,64,8,7
   6,43,63,10,74,
   45,63,10,75,44
   ,40,5,63,11,77
   ,15,98,19
110:DATA 38,8,63,1
   2,79,7,99,18,3
   6,11,63,14,98,
   17,34,14,64,14
   ,97,18,32,17,6
   4,15
120:DATA 96,18,30,
   20,65,15,95,17
   ,28,23,65,16,9
   4,17,26,25,66,
   44,24,27,67,42
   ,22,28
130:DATA 53,1,68,4
   0,20,29,53,1,6
   9,38,18,31,52,
   2,70,36,2,5,15
   ,33,51,3,71,34
   ,1,47
140:DATA 50,4,72,3
   2,1,52,74,17,1
   ,52,69,1,114,6
   ,1,51,67,2,108
   ,15,2,49,64,4,
   98,29
150:DATA 3,47,62,5
   ,87,43,4,45,60
   ,6,77,54,6,42,
   58,7,68,34,107
   ,25,7,40,56,8,
   65,34
160:DATA 108,24,9,
   37,55,8,66,30,
   109,23,10,36,5
   4,8,67,26,110,
   22,14,13,31,15
   ,53,9
170:DATA 68,23,110
   ,22,31,15,52,1
   0,69,20,110,22
   ,30,16,51,11,7
   0,20,110,22,29
   ,17,51,12
180:DATA 71,20,110
   ,22,28,18,51,1
   3,79,20,110,21
   ,27,19,51,13,7
   9,21,110,20,26
   ,20,51,14
190:DATA 80,21,110
   ,19,25,21,51,1
   4,77,24,110,19
   ,24,22,52,14,7
   3,28,109,19,23
   ,23,53,14
200:DATA 73,27,108
   ,19,23,23,54,1
   4,74,25,107,19
   ,23,23,55,14,7
   5,20,106,19,23
   ,23,56,14
210:DATA 76,16,105
   ,19,23,12,37,9
   ,57,14,80,10,1
   04,19,23,11,37
   ,9,58,14,80,10
   ,103,19
220:DATA 24,11,37,
   10,59,14,80,10
   ,102,19,25,9,3
   7,10,60,14,77,
   43,26,7,37,11,
   61,58
230:DATA 28,3,37,1
   1,62,56,37,11,
   63,54,37,11,64
   ,52,37,10,65,5
   0,37,8,66,48,3
   8,5,67
240:DATA 46,68,44,
   69,26,0,0

STATUS 1          1559
```

HAPPY BIRTHDAY

HP41C/CV

當你的朋友生日時，你會替他或她唱生日快樂，但若你很忙，而又想另他或她驚奇的話。

這樣可以叫HP－41C／CV替你服務，當他或她按下"ON"鍵後，HP－41C／CV便會爲你唱生日快樂的歌。

此程序必須具備Time－Module，否則在第76行中的一功能"ADATE"不能應用。若沒有Time－Module的用家可以用其他方法代替，將程序改變爲以下程式：

```
07 "DATE? MDY"
08 PROMPT
09 ASTO 07
75 ARCL 07
76 AOFF
```

此程序之指引爲：

（1）按XEQ "HPY"

顯示
NAME?
DATE?MDY

（2）按入你朋友的姓名跟着R／S（注意：不能超過六個字母）

（3）按入生日的日期以月，日，年方式儲入。（注意：TIME－MODULE用家則用英文儲入記憶中）

（4）HP－41C便自動將機關掉。

（5）另一用家可將HP－41C／CV "ON" 開動。

你便聽到"Happy Birthday to???"的歌。

此程序源於H.P－Key Notes；後經改善便完成"HPY"程序。

```
01 *LBL"HPY "
02 CF21
03 AON
04 "NAME?"
05 PROMPT
06 ASTO 06
07 AOFF
08 "DATE? MDY"
09 PROMPT
10 SF 11
11 OFF
12 " HAPPY"
13 ASTO 00
12 " BIRTH"
15 ASTO 02
16 "   DAY"
17 ASTO 03
18 "    TO"
19 ASTO 04
20 "   YOU"
21 ASTO 05
22 XEQ 01
23 XEQ 02
24 XEQ 01
25 XEQ 03
26 VIEW 00
27 TONE 1
28 VIEW 00
29 TONE 1
30 VIEW 02
31 TONE 8
32 VIEW 03
33 TONE 6
34 " DEAR,"
35 AVIEW
36 TONE 4
37 ARCL 06
38 AVIEW
39 TONE 8
40 TONE 2 (7)
41 TONE 2 (7)
42 VIEW 00
43 TONE 7
44 VIEW 00
45 TONE 7
46 VIEW 02
47 TONE 6
48 VIEW 03
49 TONE 4
50 XEQ 03
51 GTO 04
52*LBL 01
53 VIEW 00
54 TONE 1
55 VIEW 00
56 TONE 1
57 VIEW 02
58 TONE 2
59 VIEW 03
60 TONE 1
61 RTN
62*LBL02
63 VIEW 04
64 TONE 4
65 VIEW 05
66 TONE 3
67 RTN
68*LBL 03
69 VIEW 04
70 TONE 5
71 VIEW 05
72 TONE 4
73 RTN
74*LBL 04
75 FIX 6
76 ADATE  <- TIME
77 AVIEW
78 END.
```

SEA BATTLE II
TI58C/59

「海戰」程序是筆者兩年前的傑作，此程序是以現時學生所玩的海戰相似。只可惜正因技術不足及程序記憶太少不能讓Ti－58／59攻擊。不過「海戰」程序玩法則改了規則，以下便是規則：（用家在10×10方格內與Ti 艦隊作戰。）

（1）Ti－58C／59用家可以不停攻擊，將Ti－58C／59當作轟炸機。

（2）Ti－58C／59的船隊亦可以不停移動，令用家難於轟炸。當然Ti－58C／59亦會指供資料（code）關於Ti－58C／59艦隊消息。

當你轟炸位置後，Ti－58C／59便告知在附近八格中是否有艦隊存在。若有，Ti－58C／59顯示格內之數字。若否則顯示2．2。

※程序指引：

（1）按"E"，後Ti－58C／59便顯示"0"。

（2）跟着按"A"連同位置，Ti－58C／59顯示你攻擊多少次，跟着顯示結果。

X，X　O　Y
有幾多戰艦仍在。

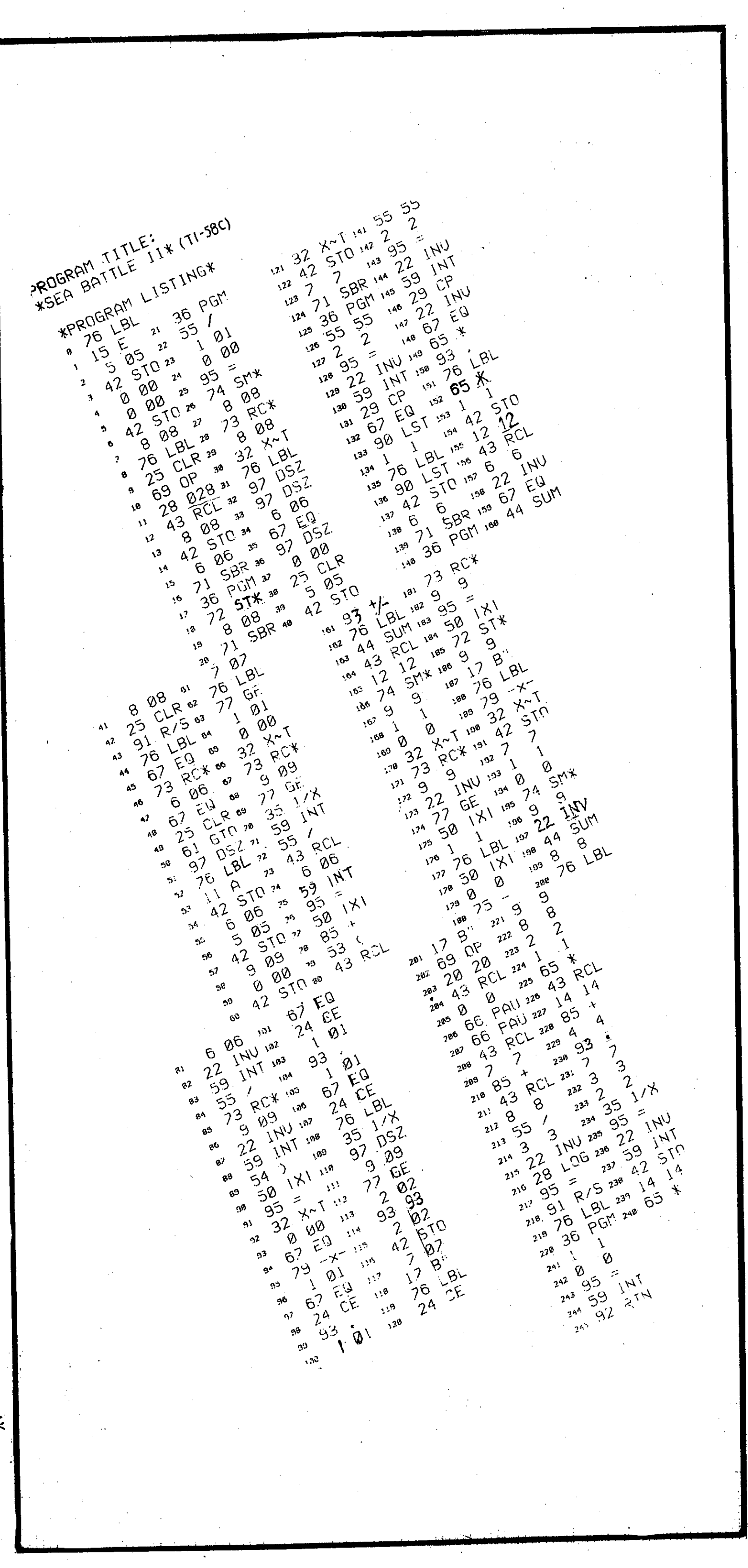
消息即Code。

現舉例示範：（見圖一）

（1）按"E"
顯示：　0

（2）按2．2跟着A
顯示：　1
1．005

前1．0提示有戰艦在1．2或3．2位置，而5顯示還有多少戰艦未擊沉。注意戰艦是可以移動，而Ti－58C／59不會提供任何資料。



```
PROGRAM TITLE:
*SEA BATTLE II* (TI-58C)

*PROGRAM LISTING*
 0  76 LBL      21  36 PGM
 1  15 E        22  55 /
 2   5 05       23   1 01
 3  42 STO      24   0 00
 4   0 00       25  95 =
 5   0 00       26  74 SM*
 6  42 STO      27   8 08
 7   8 08       28  73 RC*
 8  76 LBL      29   8 08
 9  25 CLR      30  32 X~T
10  69 OP       31  76 LBL
11  28 028      32  97 DSZ
12  43 RCL      33  97 DSZ
13   8 08       34   6 06
14  42 STO      35  67 EQ
15   6 06       36  97 DSZ
16  71 SBR      37   0 00
17  36 PGM      38  25 CLR
18  72 ST*      39   5 05
19   8 08       40  42 STO
20  71 SBR
     7 07

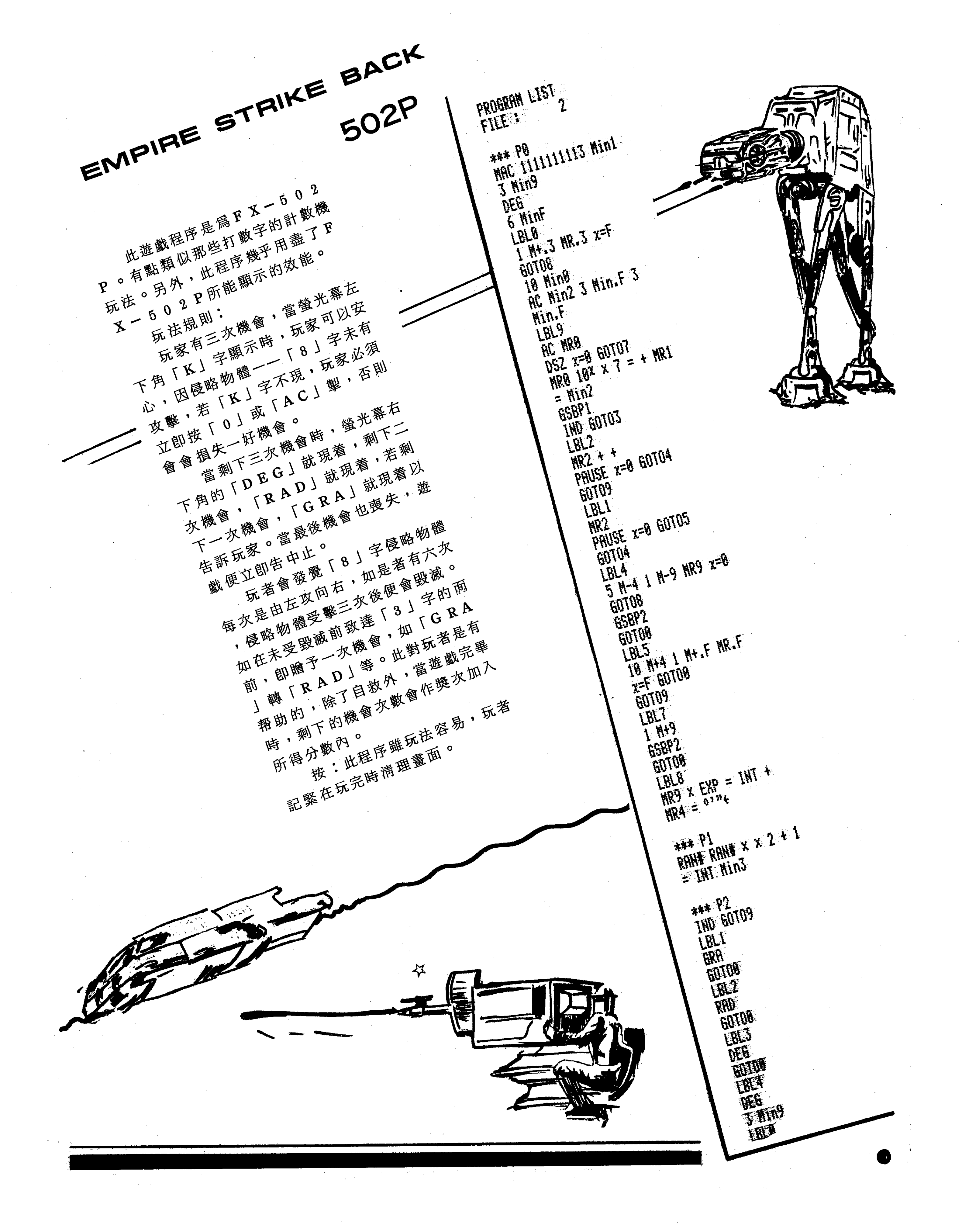
41   8 08       61  76 LBL
42  25 CLR      62  77 GE
43  91 R/S      63   1 01
44  76 LBL      64   0 00
45  67 EQ       65  32 X~T
46  73 RC*      66  73 RC*
47   6 06       67   9 09
48  67 EQ       68  77 GE
49  25 CLR      69  35 1/X
50  61 GTO      70  59 INT
51  97 DSZ      71  55 /
52  76 LBL      72  43 RCL
53  11 A        73   6 06
54  42 STO      74  59 INT
55   6 06       75  95 =
56   5 05       76  50 IXI
57  42 STO      77  85 +
58   9 09       78  53 (
59   0 00       79  43 RCL
60  42 STO      80

81   6 06      101  67 EQ
82  22 INV     102  24 CE
83  59 INT     103   1 01
84  55 /       104  93 .
85  73 RC*     105   1 01
86   9 09      106  67 EQ
87  22 INV     107  24 CE
88  59 INT     108  76 LBL
89  54 )       109  35 1/X
90  50 IXI     110  97 DSZ
91  95 =       111   9 09
92  32 X~T     112  77 GE
93   0 00      113   2 02
94  67 EQ      114  93 .
95  79 -X-     115   2 02
96   1 01      116  42 STO
97  67 EQ      117   7 07
98  24 CE      118  17 B'
99  93 .       119  76 LBL
100  1 01      120  24 CE

121  32 X~T    141  55 55
122  42 STO    142   2 2
123   7 7      143  95 =
124  71 SBR    144  22 INV
125  36 PGM    145  59 INT
126  55 55     146  29 CP
127   2 2      147  22 INV
128  95 =      148  67 EQ
129  22 INV    149  65 *
130  59 INT    150  93 .
131  29 CP     151  76 LBL
132  67 EQ     152  65 *
133  90 LST    153   1 1
134   1 1      154  42 STO
135  76 LBL    155  12 12
136  90 LST    156  43 RCL
137  42 STO    157   6 6
138   6 6      158  22 INV
139  71 SBR    159  67 EQ
140  36 PGM    160  44 SUM

161  93 +/-    181  73 RC*
162  76 LBL    182   9 9
163  44 SUM    183  95 =
164  43 RCL    184  50 IXI
165  12 12     185  72 ST*
166  74 SM*    186   9 9
167   9 9      187  17 B'
168   1 1      188  76 LBL
169   0 0      189  79 -X-
170  32 X~T    190  32 X~T
171  73 RC*    191  42 STO
172   9 9      192   7 7
173  22 INV    193   1 1
174  77 GE     194   0 0
175  50 IXI    195  74 SM*
176   1 1      196   9 9
177  76 LBL    197  22 INV
178  50 IXI    198  44 SUM
179   0 0      199   8 8
180  75 -      200  76 LBL

201  17 B'     221   9 9
202  69 OP     222   8 8
203  20 20     223   2 2
204  43 RCL    224   1 1
205   0 0      225  65 *
206  66 PAU    226  43 RCL
207  66 PAU    227  14 14
208  43 RCL    228  85 +
209   7 7      229   4 4
210  85 +      230  93 .
211  43 RCL    231   7 7
212   8 8      232   3 3
213  55 /      233   2 2
214   3 3      234  35 1/X
215  22 INV    235  95 =
216  28 LOG    236  22 INV
217  95 =      237  59 INT
218  91 R/S    238  42 STO
219  76 LBL    239  14 14
220  36 PGM    240  65 *
241   1 1
242   0 0
243  95 =
244  59 INT
245  92 RTN
```



EMPIRE STRIKE BACK

502P

此遊戲程序是爲ＦＸ－５０２Ｐ。有點類似那些打數字的計數機玩法。另外，此程序幾乎用盡了ＦＸ－５０２Ｐ所能顯示的效能。

玩法規則：

玩家有三次機會，當螢光幕左下角「Ｋ」字顯示時，玩家可以安心，因侵略物體——「８」字未有攻擊，若「Ｋ」字不現，玩家必須立即按「０」或「ＡＣ」掣，否則會會損失一好機會。

當剩下三次機會時，螢光幕右下角的「ＤＥＧ」就現着，剩下二次機會，「ＲＡＤ」就現着，若剩下一次機會，「ＧＲＡ」就現着以告訴玩家。當最後機會也喪失，遊戲便立即告中止。

玩者會發覺「８」字侵略物體每次是由左攻向右，如是者有六次，侵略物體受擊三次後便會毀滅。如在未受毀滅前致達「３」字的而前，即贈予一次機會，如「ＧＲＡ」轉「ＲＡＤ」等。此對玩者是有帮助的，除了自救外，當遊戲完畢時，剩下的機會次數會作獎次加入所得分數內。

按：此程序雖玩法容易，玩者記緊在玩完時清理畫面。

```
PROGRAM LIST
FILE :      2

*** P0
MAC 1111111113 Min1
3 Min9
DEG
6 MinF
LBL0
1 M+.3 MR.3 x=F
GOTO8
10 Min0
AC Min2 3 Min.F 3
Min.F
LBL9
AC MR0
DSZ x=0 GOTO7
MR0 10x x 7 = + MR1
= Min2
GSBP1
IND GOTO3
LBL2
MR2 + +
PAUSE x=0 GOTO4
GOTO9
LBL1
MR2
PAUSE x=0 GOTO5
GOTO4
LBL4
5 M-4 1 M-9 MR9 x=0
GOTO8
GSBP2
GOTO0
LBL5
10 M+4 1 M+.F MR.F
x=F GOTO0
GOTO9
LBL7
1 M+9
GSBP2
GOTO0
LBL8
MR9 x EXP = INT +
MR4 = °'"

*** P1
RAN# RAN# x x 2 + 1
= INT Min3

*** P2
IND GOTO9
LBL1
GRA
GOTO0
LBL2
RAD
GOTO0
LBL3
DEG
GOTO0
LBL4
DEG
3 Min9
LBL
```

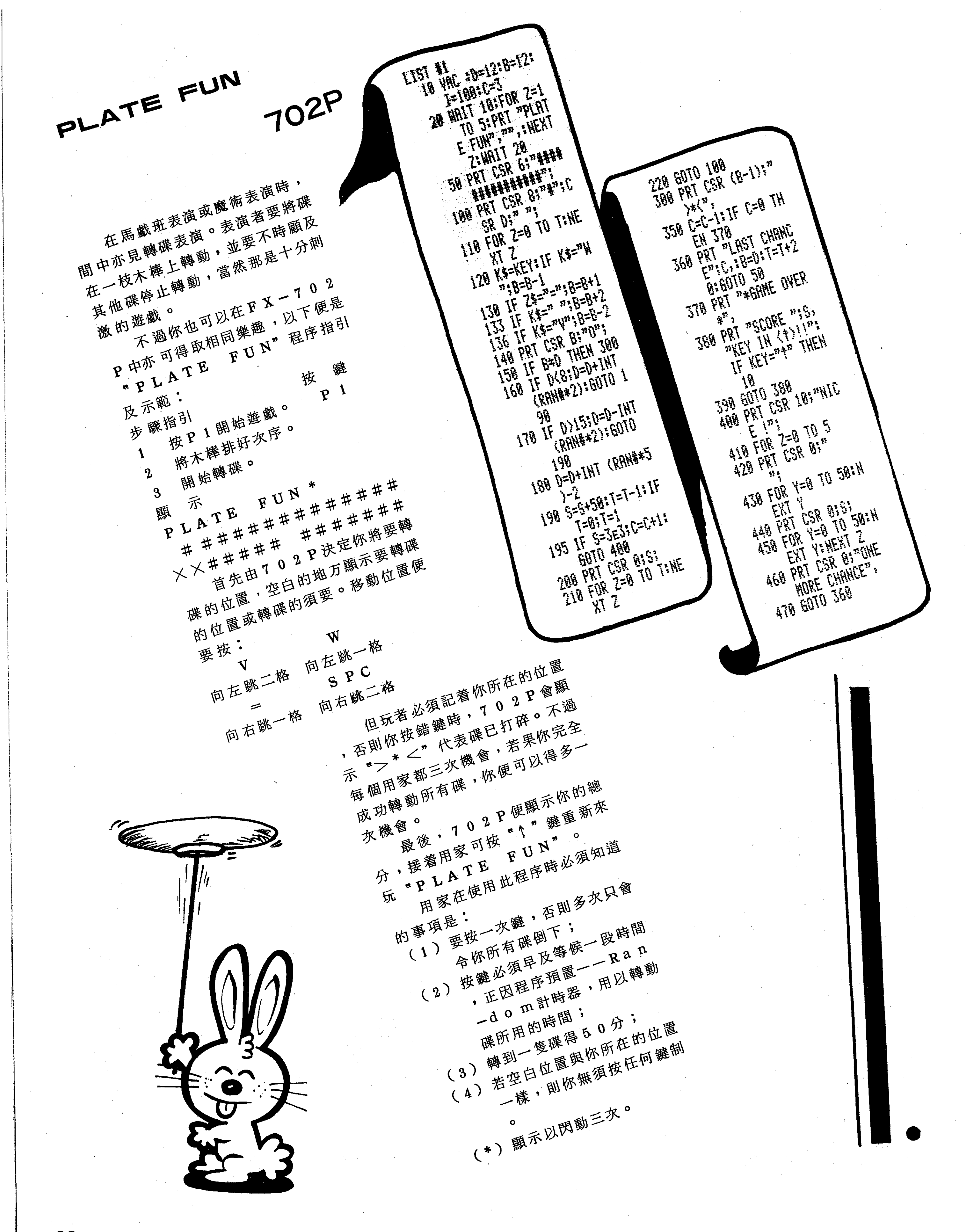


PLATE FUN 702P

在馬戲班表演或魔術表演時，間中亦見轉碟表演。表演者要將碟在一枝木棒上轉動，並要不時顧及其他碟停止轉動，當然那是十分刺激的遊戲。

不過你也可以在FX－702P中亦可得取相同樂趣，以下便是"PLATE FUN"程序指引及示範：

步驟	指引	按鍵	顯示
1	按P1開始遊戲。	P1	PLATE FUN *
2	將木棒排好次序。		##############
3	開始轉碟。		######　######
			××######

首先由702P決定你將要轉碟的位置，空白的地方顯示要轉碟的位置。移動位置便要按：

- W　向左跳一格
- V　向左跳二格
- =　向右跳一格
- SPC　向右跳二格

但玩者必須記着你所在的位置，否則你按錯鍵時，702P會顯示"＞*＜"代表碟已打碎。不過每個用家都三次機會，若果你完全成功轉動所有碟，你便可以得多一次機會。

最後，702P便顯示你的總分，接着用家可按"↑"鍵重新來玩"PLATE FUN"。

用家在使用此程序時必須知道的事項是：

（1）要按一次鍵，否則多次只會令你所有碟倒下；

（2）按鍵必須早及等候一段時間，正因程序預置——Ran-dom計時器，用以轉動碟所用的時間；

（3）轉到一隻碟得50分；

（4）若空白位置與你所在的位置一樣，則你無須按任何鍵制。

（*）顯示以閃動三次。

```
LIST #1
10 VAC :D=12:B=12:
   J=100:C=3
20 WAIT 10:FOR Z=1
   TO 5:PRT "PLAT
   E FUN","",:NEXT
   Z:WAIT 20
50 PRT CSR 6;"####
   ##########";
100 PRT CSR 8;"#";C
   SR D;" ";
110 FOR Z=0 TO T:NE
   XT Z
120 K$=KEY:IF K$="W
   ";B=B-1
130 IF Z$="=";B=B+1
133 IF K$=" ";B=B+2
136 IF K$="V";B=B-2
140 PRT CSR B;"O";
150 IF B≠D THEN 300
160 IF D<8;D=D+INT
   (RAN#*2):GOTO 1
   90
170 IF D>15;D=D-INT
   (RAN#*2):GOTO
   190
180 D=D+INT (RAN#*5
   )-2
190 S=S+50:T=T-1:IF
   T=0;T=1
195 IF S=3E3;C=C+1:
   GOTO 400
200 PRT CSR 0;S;
210 FOR Z=0 TO T:NE
   XT Z
220 GOTO 100
300 PRT CSR (B-1);"
   >*<",
350 C=C-1:IF C=0 TH
   EN 370
360 PRT "LAST CHANC
   E";C,:B=D:T=T+2
   0:GOTO 50
370 PRT "*GAME OVER
   *",
380 PRT "SCORE ";S;
   "KEY IN <↑>!!":
   IF KEY="↑" THEN
   10
390 GOTO 380
400 PRT CSR 10;"NIC
   E !";
410 FOR Z=0 TO 5
420 PRT CSR 0;"
   ";
430 FOR Y=0 TO 50:N
   EXT Y
440 PRT CSR 0;S;
450 FOR Y=0 TO 50:N
   EXT Y:NEXT Z
460 PRT CSR 0;"ONE
   MORE CHANCE",
470 GOTO 360
```

☆ BLIND DIC ☆ HP41C/CV

此程序本來是用作與另一會作賽，但後來發現另一程序較適合，所以筆者仍保留此程序一段長時間。

盲公骰程序在微型電腦中也會刊出，只不過是供ＦＸ－７０２Ｐ使用。

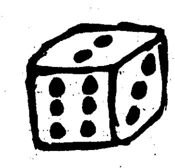
盲公骰玩法：

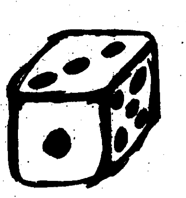
（１）首先每人拿出２１籌碼，當作彩池。
（２）每人輪次序將六個骰子打骰註：每個骰子只有一面有數字，分別由一至六刻在不同的骰子上。）將骰子面的數字加起，然後在彩池將該筆籌碼交與打骰的用家，但若果加起數字是大過彩池，打骰的用家便拿同樣數值籌碼加入彩池中，
（３）直至彩池剛好相等零，這樣最後打骰的用家更可得額外彩金，由各人拿相同數值（骰子加起的數字）給予贏家。

此ＢＤＩＣ程序玩法：

（１）按ＸＥＱ"ＢＤＩＣ"，見"ＳＥＥＤ？"便按下Ｒａ－ｎｄｏｍ－Ｓｅｅｄ，Ｒ（０≦Ｒ≦１）。
（２）跟着ＨＰ－４１Ｃ／ＣＶ便問你有幾多人玩，那樣按入數字。當用家發現"ＳＩＺ－Ｅ＝＞×××"，即ＨＰ－４１Ｃ／ＣＶ的記憶系統太少，那時用家便要ＳＩＺ－Ｅ　×××。
（３）按着４１Ｃ／ＣＶ問每一用家之姓名，（六個字母）
（４）ＨＰ－４１Ｃ／ＣＶ便跟着開始遊戲。
（５）４１Ｃ／ＣＶ顯示用家姓名及彩池，用家可以按下一數值爲幸運號碼。直至遊戲完畢，ＨＰ－４１Ｃ／ＣＶ便顯示各用家所得的籌碼。

```
01*LBL "BDIC"
"SEED?" PROMPT ABS
ST+ 00
06*LBL 08
"PLAYER? P=)2"
PROMPT CF 05 CF 29
FIX 0 STO 02 1.001
* STO 04 2 * 5 +
SF 25 RCL IND X
FS? C 25 GTO 00
"SIZE =)" ARCL X
PROMPT
27*LBL 00
5.005 RCL 02 + STO 01
RCL 04 X()Y + LASTX
-21 "NAME?" AON
39*LBL 01
PROMPT ASTO IND Y
STO IND Z DSE Z
"NEXT?" DSE Y GTO 01
AOFF "EACH PAT M:21"
AVIEW ABS ST* 02
52*LBL 02
RCL 01 STO 05
55*LBL 03
CLX STO 03 CLA
ARCL IND 05 "├ M: "
ARCL 02 TONE 9 TONE 7
PROMPT ABS SQRT FRC
ST+ 00 " GOOD LUCK" 5
RCL 02 X(=Y? AVIEW
11.017
75*LBL 12
CF IND X ISG X GTO 12
CLA 6 .3
82*LBL 04
XEQ 09 X)Y? GTO 10
SF 05
87*LBL 11
RDN XEQ 09 6 * 1 +
INT 11 X()Y +
FS? IND X GTO 11
SF IND X CLX LASTX
103*LBL 10
FC? C 05 CLX TONE IND X
ST+ 03 ARCL X RDN
"├" AVIEW DSE Y
GTO 04 RCL 02 RCL 03
X)Y? CHS ENTER↑ X=0?
GTO 14 X)0? GTO 13
X() Z + 21 / .25 /
27 + GTO 15
132*LBL 14
29 GTO 15
REM
135*LBL 13
X() Z 21 X(Y? X()Y
RDN / .125 / 31 +
146*LBL 15
XEQ 09 2 * +
XEQ IND X AVIEW TONE 9
PSE RDN ST- 02 RCL 05
RCL 04 + X()Y
ST+ IND Y X)0?
"YOU GAIN M:" X(0?
"YOU LOSS M:" ABS X≠0
ARCL X AVIEW BEEP
DSE 05 GTO 03 GTO 02
174*LBL 09
9821 RCL 00 * .211327
+ FRC STO 00 RTN
183*LBL 39
184*LBL 40
"-CONGRAT" "├ULATION-"
TONE 9 AVIEW TONE 7
PSE "YOU MAKE IT"
AVIEW AVIEW BEEP RDN
RCL 01 RCL 04 + X()Y
200*LBL 06
ST- IND Y DSE Y GTO 06
RCL 04 * ST+ IND Z
RCL 04 RCL 01 + LASTX
211*LBL 07
CLA ARCL IND X "├ M:"
ARCL IND Y AVIEW
TONE 5 TONE 5 TONE 5
TONE 4 PSE DSE Y X=0?
DSE X GTO 07 STOP
GTO 08
228*LBL 22
"HELPLESS" RTN
231*LBL 21
"TOO BAD" RTN
234*LBL 24
"HOPLESS" RTN
237*LBL 23
"HO PITY" RTN
240*LBL 26
"POOR" RTN
243*LBL 25
"SAD?" RTN
246*LBL 28
"YEHHH..." RTN
249*LBL 27
"OHHH..." RTN
252*LBL 30
"GAIN NOTHING" RTN
255*LBL 29
"NO CHANGE" RTN
258*LBL 31
"TOO LESS" RTN
261*LBL 32
"SO LITTLE" RTN
264*LBL 33
"NOT BAD" RTN
267*LBL 34
"LUCKILY" RTN
270*LBL 35
"GOOD" RTN
273*LBL 36
"GREAT PRIZE" RTN
276*LBL 37
"FINE" RTN
279*LBL 38
"TOO PERFECT" RTN END
```







考你記憶！（FX－702P ONLY）

「考你記憶」的遊戲節目也可能見過，當你拿起一圖型時，你便要到處找另一相同圖型位置。若果你所拿起的圖型與先前那個不同，你便立即放回那兩個圖型的位置，重新來一次找另一對。但若找對，便可拿出來，這樣便減少再重找的機會。當然一個人玩的趣味不多，所以筆者再應用程序可借兩人參用。

以下就是「考你考憶！」程序的指引：

（1）按"P0" 跟着顯示"PLAYER：1 OR 2?"

（2）若一用家使用，則按"1"；但若兩用家一起玩，則按"2"，並跟着按入首用家的名字；當702P顯示"1ST NAME?"接着"2ND NAME?"，另一用家的名字再儲入702P記憶系統中。（記着：人名不能超過七個字母。）

（3）跟着702P顯示"RANGING"；將一對由2至9及A的字排好不同次序。

（4）用家便見"？？？？？？？？？？？？？？？？？？？？？？？？*？"，"*"是用家的位置或第一個用家之位置；而"+"是第二個用家的位置。

（5）用家按以下鍵制有不同的功能：
V：開牌　4：向左移
6：向右移

（6）用家按到不同的字，702P便會暗藏起來，用家必須記着那個位置的字。但相同的話，702P便會一直將字顯示。

（7）直至各字母全部顯示，702P便會計分數，若一人玩則計算你的記憶效力分數。若兩人玩則作比較。

各用家可試自己的記憶力有幾多？

```
LIST #0
 10 VAC :WAIT 0:$="
    12A23456789":H$
    ="*":S$="+"
 14 L=1:R=.5:T=18:P
    =19:PRT "PLAYER
    : 1 OR 2?"
 15 Z$=KEY:IF Z$=""
    THEN 15
 16 FOR I=1 TO 2:IF
    Z$=MID(I,1) TH
    EN 40-I*10
 17 NEXT I:GOTO 15
 20 INP "1ST NAME",
    F$,"2ND NAME",G
    $:Q=1
 30 PRT "RANGING":$
    =MID(3)
 35 FOR I=0 TO 8:A$
    (I)=MID(I+1,1):
    A$(I+9)=A$(I):N
    EXT I
 40 FOR I=0 TO 17:C
    =INT (RAN#*18):
    B$=A$(I):A$(I)=
    A$(C):A$(C)=B$
 45 NEXT I
 50 FOR I=1 TO 18:P
    RT "?";:NEXT I:
    GOTO 150
 60 Z$=KEY:IF Z$="4
    ";L=-1:GOTO 100
 70 IF Z$="6";L=1:G
    OTO 100
 80 IF Z$="V";GOTO
    (170-X*110)
 90 X=0:GOTO 60
100 PRT CSR M;"?";
110 M=M+L:IF M=R TH
    EN 110
120 IF M<0;M=18:GOT
    O 110
130 IF M>17;M=-1:GO
    TO 110
140 IF A$(M)="" THE
    N 110
150 PRT CSR M;H$;:G
    OTO 60
170 PRT CSR M;A$(M)
    ;:W=W+1:X=1
190 IF Y=0;Y=1:R=M:
    GOTO 110
200 Y=0:IF A$(R)=A$
    (M) THEN 280
220 FOR B=1 TO 100:
    NEXT B
240 IF Q=1;U$=H$:H$
    =S$:S$=U$:O=T:T
    =P:P=O
250 PRT CSR R;"?";:
    R=.5:GOTO 150
280 A(T)=A(T)+1:IF
    A(T)+A(P)=9 THE
    N 320
310 A$(M)="":A$(R)=
    "":R=.5:GOTO 11
    0
320 PRT :IF Q=1;PRT
    F$;A(18)*2;":"
    ;G$;A(19)*2:END

330 PRT W;" TIMES";
    INT (18/W*100+.
    5);" MARKS":END
```

記

702P



ACTION



當你首次接觸你的袋裝電腦時，你會發覺對鍵制排位次序非常陌生。但現在只要玩"Action Game"，你便很容易記着鍵制排位。

以下便是"Action Game"的指引：

步驟		
1	按P2	ACTION-GAME READY
2	當你準備好，便要按"Y"鍵，跟着"EXE"。便開始"Action Game"。	

各用家一見由英文字母"A-N"閃現後，立即按回那英文字之鍵，若你按對的話，你的袋裝電腦便顯示"CORRECT！××"（註：××是你所得的分數"。）否則你的電腦顯示"WRONG ANS=Y"（註：Y是那個要按的鍵制。）

當你被測試二十次後，電腦會告知你的總分。

```
LIST #2
10 VAC :$="ABCDEFG
   HIJKLMN"
20 WAIT 9:PRT "ACT
   ION-GAME":WAIT
   1
25 INP "READY",U$:
   IF U$≠"Y" THEN
   25
30 FOR I=1 TO 20
40 N=INT (RAN#*14+
   1)
50 S=INT (RAN#*11)
60 A$=MID(N,1):K$=
   ""
70 PRT CSR S;A$:PR
   T ""
80 FOR J=1 TO 50 S
   TEP 2
90 K$=KEY:IF K$≠""
   THEN 110
100 NEXT J
110 IF K$=A$;M=M+55
    -J:PRT "CORRECT
    !";55-J
120 IF A$≠K$;PRT "W
    RONG ANS= ";A$
130 NEXT I
140 PRT "SCORE=";M
150 END
```

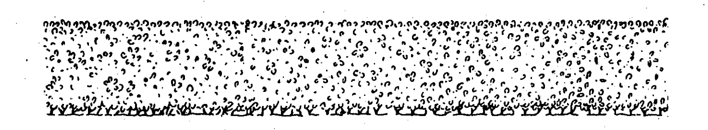
步驟	指示	按鍵	顯示
1	按"P2"	P2	ACTION-GAME READY
2	準備好，便按Y	Y EXE	A≪閃
3	按回答案 跟着另一字母	A	CORRECT! 28 J
4	按回答案 ⋮ 總分顯示	I ⋮	WRONG ANS=J ⋮ SCORE=328

註：英文字母顯示在十二個顯示位中閃現一次。各用家試嘗試破紀錄，對你的電腦按鍵排位次序有更深刻記憶！

IE 80

PC Journal

PC



我們新的會期（８２～８３）年度已正式開始，同時趁著新會期的開始，亦把會名改稱爲ＰＣ　Ｃｌ—ｕｂ，而由於得到現代電子出版社的協助，本會得以把名爲ＰＣ　Ｊｏｕｒｎａｌ的主要通訊的刊物，改刊於這一本全新的雜誌——ｉ．ｅ．８０——之上，從而令更多讀者悉知本會的存在。

過去曾有不少熱心的讀者，來信查詢本會的資料，包括組織的結構和工作的情況，但奈何由於本會人手不足，以致未能及早囘覆，爲此本會同人謹向這些熱心的讀者致萬二分之抱歉；現在我們打算在此簡略地介紹一下，以答覆讀者的來信。

起初，本會和其他的類似組織一樣，都是由一群熱愛這類玩意的人組織而成，最初只是幾個學生在一起湊湊熱鬧而已。但經過多年的發展，現在業已成爲一個有頗多會員的組織了。在結構上是分爲委員和會員，一切的會務都是由委員負責的，而大部份的委員都是創立本會的人。在消息聯絡方面，主要是依賴通訊刊物ＰＣ　Ｊｏｕｒｎａｌ，和書信的往來，通過這些工具，足以將委員和會員連繫起來。本會的主席是Ｂｅｎｎｙ　Ｗｏｎｇ先生，而負責對外工作和消息傳遞的是Ｗｏｎｇ　Ｃｈｏｉ，當然並不是只有兩位委員，但他們兩人是主要的代表。

由於新的年度（８２～８３）開始，委員會亦已製定新一年度的行政方針；在消息傳遞方面：我們是以ＰＣ　Ｊｏｕｒｎａｌ爲主，它的內容已在前文交待了，雖然ＰＣ　Ｊｏｕｒｎａｌ是主要的消息傳遞工具，但它並不代表所有，對於一些突發性的消息和重要的通告等，都會以個別發信通知，以務求達到消息最快的宗旨；此外本會計劃可能租借地方與各會員共聚，以便更直接地交流意見和資料。

除了計劃與會員聚會之外，我們的Ｐｒｏｇｒａ—ｍ　Ｌｉｂａｒｙ會繼續提供程序於會員，但單方面的供應實在太沒有意思，爲了能令各會員可以發揮自己的才能，我們鼓勵會員自行設計程序，完成之後可以把它寄來，倘若適合的話，我們會替程序作出評論寄囘的。

通常每一位會員要求的軟件程序，是各有不同的，如果有足夠的資料，會員可以要求本會協助編寫，若有必要的話，會徵求其它會員協助編寫，這一類的要求本會必定盡力帮助的。

所有刊出的程序，若會員認爲有可以改善的地方，歡迎會員提出批評和指教，以期望達到技術交流的效果。

＊（所有刊出的程序和文章批評，本會將於以獎勵。）

各位會員亦曾經提出一些意見，「買賣交換」計算機消息站，便是徇衆要求而設立的，各會員可以把想買、賣或交換的機型和配件，購入時間等有關的資料，連同通訊地址一併寄來，（奉勸請勿填寫電話號碼，以避免無聊的人士所滋擾。）會方面將一切可以刊登的資料刊出。有意洽商者，可以藉著刊出的資料和通訊地址和對方聯絡，本會並不會作中間的介紹人，（正所謂唔做中、唔做保、唔做媒人三代好是也。）一切的交易，與本會絲毫沒有關係，希望大家注意。

至於其他的活動，如程序比賽等，由於在細節方面仍未議定，容待稍後公佈，倘若有甚麼建議，各會員盡管來信，我們必定詳加考慮。

若讀者有意成爲ＰＣ　Ｃｌｕｂ成員之一，讀者只要寫信寄來本會，本會盡快寄出申請表。



什麼叫做P．C．呢？你知道它有幾多呢！

P．C．全名就是〝POCKET COMPU－TER〞（袋裝式電腦）或〝PROGRAMMA－BLE CALCULATOR〞（程序式計數機）。衆所週知P．C．大多是手提式，但是它的演變却是千變萬化。從第一部面世的計數機距今不過短短的幾十年時間，而P．C．已經能與大型電腦比美，由此可見科技一日千里，進步神速！

在香港PC市場上現時最常見的牌子爲：Casio（金石），Hewlett－Packard（惠普）；Sharp（聲寶）及Texas Instrument（德州）。現在來一次PC巡禮吧！

（1）Casio（金石）計數機最多花款，令人眼花瞭亂。憑我的記憶中首先在港發售之P．C．爲：Fx－1；Fx－201P及Fx－202P。Fx－201P與Fx－202P之差別只是後者爲Continovs Memory。而Fx－1則多一磁咭機（Card Reader）。而磁咭的厚度是十分厚的。此類型P．C．的程序語言都與現時流行的Basic極爲相似。過了幾年後，袋裝型Fx－5019／502P推出發售，速度相當快而慳電，被大部份P．C．用家喜用。但Fx－601P／602P，出現後令Fx－5019／502P漸漸隱退，它的外型及功能都與Fx－5019／502P大同小異。但不到數月Fx－702P亦推出，而同時期亦有小型P．C．例如：Fx－3500P／1800P／2700P／3600P等等。近期Casio新出有Fx－801P，PB－100＊及Fx－700P＊，都令PC用家有更多選擇。

（＊PB－100及Fx－700P可提駁FA－3。）

（2）Hewlett Packard（惠普）較爲少人知道的一間公司，其實它在美國最有名的公司之一，亦是最早生產P．C．其中之一。HP的P．C．以RPN爲著名，早期時代P．C．機有HP－19C，HP－29C，HP－65等。在本港現有不少專業人士如路政部都採用或擁有一部至兩部HP－67，HP－97或HP－41C／CV。HP－67外型類似Ti－59，而HP－97與HP－67功能一樣，不過它附設印刷機吧。

至於HP－41C／CV，各位對它一定不會陌生，PCCC曾經在微型電腦專刊談及HP－41C／CV至HP－IL，日後PC Jouma仍會陸續介紹更多有關HP－75C，HP－41C／CV及HP－IL資料。

近期HP新機爲HP－10C，HP－15C，HP－16C及HP－75C，各位請在下期見〝考試P．C．之霸——HP－15C〞一文有所介紹HP－15C，正由於PC Journal稿多，HP－16C及HP－75C都會在日後必會爲各位詳細介紹。（PC會員可翻閱PCCC Newsle－tter V3N1新產品一欄。）

（3）Sharp（聲寶）近期威震四方，PC－1500系列令各P．C．用家感到嘆息！現代科技日新月異，難怪使一些P．C．用家不知如何選擇P．C．才好。PC－1500在美國被人稱呼爲R－adio Shack PC－2，而PC－1就是PC－1211。而PC－1500的配件已增至CE－159（8K Continously R－AM），而CE－158即介面RS－232C將快在港出售，到時對PC－1500用家必定是一個大喜訊，而PC Journal亦會詳盡介紹PC－1500的ROM。

（4）Texas Instrument（德州）——初期PC機型爲：SR－52；Ti－55及Ti－57等。在70年代曾轟動一時之Ti－58／58C／59令PC用家初嘗Solid St－ate Softeware TM（固體軟件）或Modules的好處。

現在仍有一大部份PC用家都仍然是Ti－58C／59擁護者，但一些秘密是他們不知道的。PC Club稱呼那些新指令爲〝Ti－58C／59 Synthetic Programming〞，有部份亦在近幾期微型電腦專刊中刊出，而PC Journal亦陸續有所介紹。Ti近期推出Ti－88，詳細資料請見USPC一欄。

從上述概括介紹，各位都對本港PC市場有所認識，至於其類型及功能都有待日後再談吧！



今年五月廿六日，Ti －58C／59用家一直夢未以求之全新Ti －88已經誕生，這是Ti 一擁護者，在HP－41C／CV面世之後所期望的結晶品，希望特此取回昔日的雄風。

Ti －88外型與Ti －55Ⅱ相似，LCD－Dot matrix顯示版與水平形成15，而按鍵則與Ti －58C／59之排位次序相同，全新Ti －88給人一種新穎的感覺！

現在帶各位參觀Ti －88各項性能；

(a) LCD顯示：16個5×7點陣

(b) 字母顯示：128種另有一新顯示設計，可將兩個字母串連一起顯示在5×7點陣，例如PH可以顯示成PH。

(c) 時　間　：內置，可以改變成12或24進制，並有日子記憶系統，例如16：19：13 TUE－20，準確程度爲±0.1MS。

(d) 程　序　：1160步（最多可以到4488步）；145記憶系統（最多可以到561個）；56個標旗；
87個OP功能；
26個英文母字標記（Label）；
100個數字標記；
10個用家標記；等

(e) 聲　音　：半秒響鬧。

(f) 配件港口：兩個固體軟件，Ti 亦生產一新Memary Modvle 提供更多記憶。而這新Memory Modvle可作ROM用，正因它可以脫離Ti －88，獨自維持五年時間。
而Ti －58C／59的固體軟件，Ti －88亦可同時享有。

(g) 其他配件：(1) PC－800 Printer印刷機，性能與現時PC－100C／A相同。
(2) CA－800 Cassette Interface，性能與Casio廠生產的FA－1，FA－2及FA－3相同。

其他功能與現時Ti －58C／59相似，當然Ti －88也有英文顯示及英文字母控制。其他詳細資料將在Ti －88 Review中再替各位解答。

Ti －88前排五個鍵掣分別爲“YES”；“NO”，“UNK”；“ENT”及“CONT”，用來答覆Ti －88固體軟件程序的問題。當你將開關鍵按至SP（System Programming）時，Ti －88立即顯示“MAY I HELP YOU?”，你的答覆便按那五鍵掣，Ti －88便會爲你處理有關問題。

New AOS for Ti －88

最近Texas Instrument（簡稱T.I.），最新生產之Ti －88，便應用全新AOS，全新AOS的設計是以數學之邏輯爲根基。

現時計數機大都應用AOS（Algebra Operation System）*或RPN（Reserve Polish Notation）。但此兩種仍與計算時之邏輯有所不同，例如：計算Sin 30°時，計數機便要按30°Sin，以序顚倒。

但在Ti －88便要按Sin先，後30°。再按加多一鍵＝，答案便會顯示在螢光幕上。

雖然按多一鍵，但是很容易明白。對普通初學者，更容易接受。而Ti －88的AOS另一突出之處在於（UP）及）Dn）之鍵，試計算3↑〔2↑（1＋3）〕＝？

Ti －58C／59要按3y^x（2y^x（1＋3））＝，但在Ti －88則按3〔Up〕2〔Dp〕1＋3〔Dn〕〔Dn〕便可。

全新AOS使PC用家更易處理數學問題。

註*：AOS亦可叫作APO。



很多朋友都聽過RPN這個名字，它是一種邏輯系統，可是對於它的工作情況却不甚了解，爲此我們這次撰寫一篇有關RPN的簡介文章，希望對大家在認識RPN方面有所帮助。就所知來說，採用RPN的計算機，只有HEWLETT—PACKARD（HP）廠所出產而已。

HP機的運算邏輯是建立於一種稱爲Polis—h—Notation的數學邏輯之上，這種邏輯系統是由一位波蘭的邏輯學家Jantukasie—wicz（1878—1956）所建立的。

在計算一條數式時，一般的代數表示法是把運算符號放在兩個變量之間，而Polish—Nota—tion是把運算符號放在兩個變量之「前」。爲了方便在計算機上的操作，HP把運算符號放在兩個變量之「後」。這就是稱爲RPN（Reverse Polish Notation）的原因了。

要使到RPN在計算複雜的數式時，能夠不需用括號而簡單地計算出答案，便要利用到AMS（Au—tomatic Memory Stack）和它的輔助指令，包括ENTER，R↑，R↓，x y等。

Stack是由四個記憶單位組成的，我們分別稱它們爲X、Y、Z和T記憶，X記憶是顯示在面版上的，換句話說，若你按下任何數字在顯示版上，那些數字即已輸入X記憶裏了。

AMS簡化圖

T	t	
Z	z	
Y	y	
X	x	顯示在面版上

*我們用x，y，z，t 代表在X，Y，Z，T記憶內的數字。

ENTER這個指令是用來分隔輸入的數，使數字可以輸入Stack內的每一個記憶。方才說過RPN邏輯是把運算符號放在變量之後，現在有了EN—TER後便可以試行計算了，例如要計算2＋3，在數式是這樣的2＋3＝，而RPN則是2ENTE—R3＋，先輸入2，3兩個變量，然後才輸入＋號。

若有任何數字輸內，或因爲計算而出現的答案，都出現在顯示版上（X記憶），則會引起Stack內記憶的變化，計有三種情形：上升，不變和下降，以下是其中的例子。

上升

		失去
T	t ↗	z
Z	z ↗	y
Y	y ↗	x
X	x ↗	π

輸入π（pi）

不變

T	t →	r
Z	z →	z
Y	y →	y
X	x →	x^2

輸入x^2指令

上降

T	t →	t
Z	z ↘	t
Y	y ↘	z
X	x →	x＋y

輸入
＋號

*下降時，T記憶內數字不變，亦同時下降到Z記憶內。

我們可以用2＋3作爲例，看看Stack內四個記憶的工作情形。（灰色格表示若有任何數輸入，則格內的數字將被除去。）

T	t	z	y	y	y
Z	z	y	x	x	y
Y	y	x	2	2	x
X	x	2	2	3	2＋3
輸入		2	ENTER	3	＋

當然，倘若要計算2＋2時，則只需要按2EN—TER＋而已了。至於R↑（轉上），R↓（轉落）和x y（X，Y交換）的工作情形，可以看下圖便知曉了。

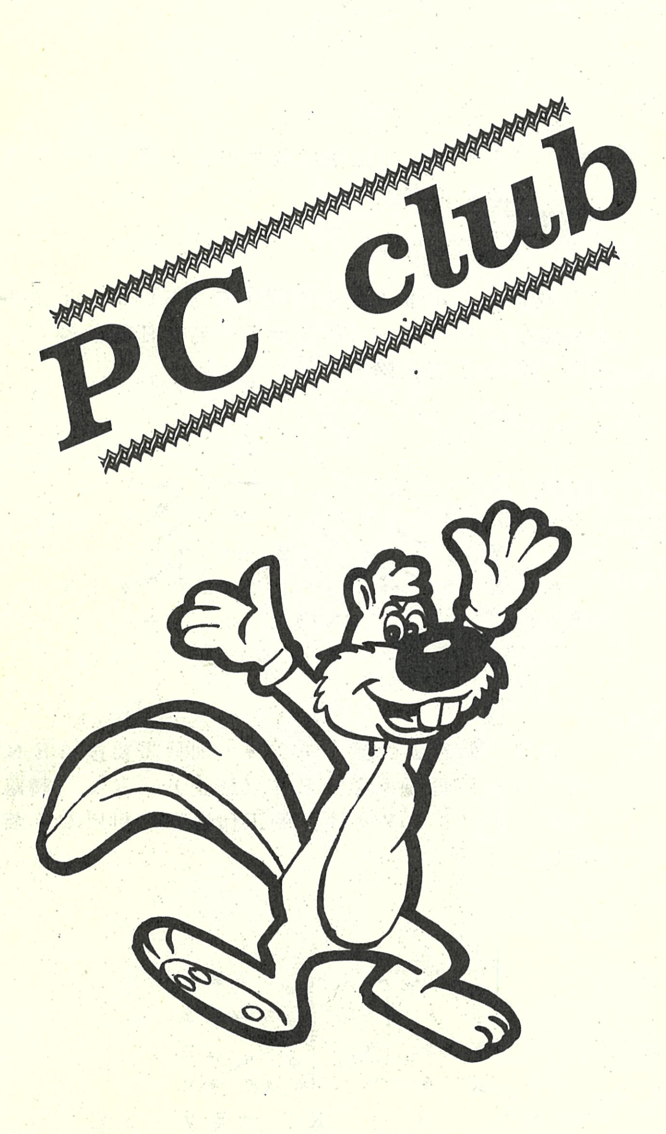
T	t	x	t	t
Z	z	t	z	z
Y	y	z	y	x
X	x	y	x	y
輸入		R↓	R↑	x⇄y

看了上圖相信對於R↓，R↑和x　y 必定有所了解。但是似乎RPN和其它的運算邏輯沒有甚麽分別，例如2＋3＝和2ENTER3＋同樣要按4次鍵，但在計算複雜的時候，情況便不同了，例如計算3〔4＋5（6＋7）〕，一般計算機是這樣輸入：6＋7＝×5＋4＝×3＝共12鍵，但用RPN便不同了；6ENTER7＋5×4＋3×共10鍵，

由於有了Stack的帮助，使到RPN的運算和編寫程序方面，省了不少的空間和時間，RPN表面上似乎是很繁複，但熟習後，便會感到比一般程序計算機更容易編寫程序。

當然RPN並不只是方才那麽少少，但一般來說，最基本的，都已經提過了，但篇幅所限，只提到這裏，希望大家對它有一個更淸楚的印象。

順帶一提，在不同型的HP機，是有不同的RPN的（當然上面所說是一般共通的）。例如在HP－41C／CV裏Stack除了可以輸入數字外，還可以輸入字母和符號，而且可以任意指定輸入Sta－ck內的任何一個記憶！而在HP－15C裏，當採用複數計算時，除了原有的實數Stack外，另外自動加上虛數Stack，並列而同時工作，……還有其它的個別情形，在這裏我們不再介紹了。●



PC Journal是由PC Club負責的一個專欄，PC Club這個名字，對各位讀者都似乎十分陌生，所以爲對讀者有所交待。現在簡略地介紹給各位讀者認識。

PC Club之前身就是PCCC，相信各位也曾在「現代電子」雜誌中見過。由於PCCC之會員激增，工作繁務對昔日之PCCC已不能應付，加上各種壓力，這是必要將PCCC重組來一次新改革，帶來新的面貌！

新的PC Club已於十二月一日開始服務，替各位P.C. *會員交換消息及程序。，**至於PC Club與各會員之通訊，是以PC Jou－rnal爲主。所以PC Club同人特此多謝「I.E. 雜誌社」能夠借出數版作爲PC C－lub之通訊。

我想各位仍然對PC Journal有很多疑問，不如概括地介紹PC Journal之內容：

1500 Notes——介紹PC－1500結構爲主，例如Ra－m及Rom等。

USP——專欄介紹美國P.C.，並以Ti－58C／59及HP－41C／CV Synthetic Program－ming爲主；間中亦提及其他P.C.怪異資料，如Ti－57怎樣變成Ti－57C。

MODE——提供Fx系列之資料及程序剖釋。

而每次都有一個特別題材介紹有關P.C.之資料，正如今期之〝RPN簡介〞，風格次次不同，令讀者大開眼界！

PC Club委員會
一九八二年十二月二日

PC JOURNAL 預告

i)EXAM-USE-PC:HP15C

ii)PC-1500 ADDRESS CODE 大揭秘
CSIZE 50 之趣味

iii)TI-58/59 SYNTHETIC PROGRAMMING

MICRO 83 快將出版



現代電子　好勁　完美的吸引
全速的改進

現代電子　好勁　完美的吸引
全速的改進